



Universidad
Carlos III de Madrid

ESCUELA POLITÉCNICA SUPERIOR

INGENIERÍA TÉCNICA INDUSTRIAL, ESP. ELECTRICIDAD

PROYECTO FIN DE CARRERA

HERRAMIENTA PARA LA TRADUCCIÓN ASISTIDA EN LA INDUSTRIA AUDIOVISUAL

Autora: Raquel Mejías Moreno
Tutor: Manuel Velasco de Diego
Director: Diego Bodas Sagi

SEPTIEMBRE DE 2010

*A mis tutores, por su orientación, su apoyo y su confianza
en que podría sacar este proyecto adelante.*

*A mis compañeros, por ayudarme a comprender
el auténtico valor de esa palabra.*

*A mi familia, por estar siempre ahí.
En especial a mi madre, que siempre da tanto sin pedir nada a cambio.*

*Y a mi novio, por su paciencia, por su eterna sonrisa
y por ser mi brújula cuando pierdo el rumbo.*



Índice de contenido

1.	Introducción	12
1.1	Motivación del proyecto	12
1.2	Objetivos	14
1.3	Estructura del documento	15
2.	Estado de la cuestión	17
2.1	Traducción asistida por ordenador.....	17
2.1.1	Traducción automática y traducción asistida por ordenador	17
2.1.2	Diccionarios electrónicos y sistemas de gestión de bases de datos terminológicas	18
2.1.3	Herramientas de análisis de corpus.....	19
2.1.4	Memorias de traducción	20
2.2	Traducción audiovisual	21
2.2.1	Aspectos generales	21
2.2.2	Subtitulación.....	22
2.2.3	Práctica profesional	23
2.2.3.1	Proceso de subtitulación.....	23
2.2.3.2	Limitaciones espacio-temporales.....	24
2.3	Herramientas existentes para la traducción de subtítulos	25
2.3.1	MacCaption	26
2.3.2	EZTitles	27
2.3.3	PoliScript 3000	28
2.3.4	WinCaps.....	29
2.3.5	Herramientas semiprofesionales.....	29
2.3.5.1	Subbits	30
2.3.5.2	Subtitle Translation Wizard	30
2.3.5.3	Subtitle Workshop.....	31
2.3.5.4	dotSUB	32



2.3.6 Comparativa entre las herramientas existentes y los objetivos planteados	33
3. Representación del conocimiento.....	35
3.1 Introducción.....	35
3.2 RDF(S) y ontologías	37
3.2.1 Resource Description Framework (RDF)	37
3.2.2 Serialización: Notation 3	42
3.2.3 Ontologías	44
3.2.4 RDF Schema.....	48
3.2.5 SPARQL.....	51
4. Planificación del proyecto.....	53
4.1 Tareas a realizar.....	53
4.2 Diagrama de Gantt	54
4.3 Presupuesto	57
4.3.1 Desglose del proyecto en fases.....	57
4.3.2 Gastos de personal	57
4.3.3 Gastos directos materiales	59
4.3.4 Resumen del presupuesto.....	59
5. Análisis del problema	61
5.1 Planteamiento del problema	61
5.2 Requisitos fijados	61
5.2.1 Requisitos funcionales.....	62
5.2.2 Requisitos no funcionales.....	69
5.3 Especificación de casos de uso.....	71
5.3.1 Escenario 1: Búsqueda de términos en el diccionario.....	72
5.3.2 Escenario 2: Modificación del diccionario	74
5.3.3 Escenario 3: Traducción de subtítulos	77
6. Diseño de la solución	82
6.1 Arquitectura del sistema	82
6.2 Componentes del sistema	83



6.2.1	Interfaz gráfica de usuario	83
6.2.1.1	Pantalla contenedora MDI (MDIParentForm)	83
6.2.1.2	Pantalla de consulta del diccionario (DictionaryForm)	84
6.2.1.3	Pantalla de edición del diccionario (EditEntryForm)	85
6.2.1.4	Pantalla de traducción de subtítulos (SubtitleForm)	85
6.2.1.5	Otras clases y pantallas	86
6.2.2	Presentación	87
6.2.3	Negocio	89
6.2.4	Servicio	94
6.2.5	Acceso a datos	94
6.3	Diagrama de paquetes	96
6.4	Repositorio terminológico	96
6.5	Diseño de la base de datos	101
7.	Implementación	102
7.1	Plataforma de desarrollo .Net	102
7.2	Codificación	102
7.2.1	Interfaz gráfica de usuario	103
7.2.2	Presentación	110
7.2.3	Negocio	112
7.2.4	Servicio	115
7.2.5	Acceso a datos	116
8.	Evaluación de la herramienta	119
9.	Trabajos futuros y conclusiones	121
9.1	Trabajos futuros	121
9.2	Conclusiones	122
10.	Bibliografía	124
11.	Apéndices	128
11.1	Apéndice A: Consultas SPARQL	128
11.2	Apéndice B: Manual de usuario	131



11.2.1 Edición del diccionario	132
11.2.1.1 Nueva entrada	132
11.2.1.2 Modificar entrada	133
11.2.1.3 Borrar entrada.....	134
11.2.1.4 Navegación entre últimas consultas.....	134
11.2.2 Consulta del diccionario.....	134
11.2.2.1 Búsqueda no segura	135
11.2.3 Traducción de subtítulos.....	135
11.2.3.1 Cargar archivos de subtitulación	136
11.2.3.2 Editar subtítulos	137
11.2.3.3 Información de los subtítulos.....	137
11.2.3.4 Ir a subtítulos determinados	138
11.2.3.5 Guardar y cerrar archivos de subtítulos.....	138
11.2.3.6 Importar y exportar subtítulos a otros formatos	139
11.2.3.7 Cargar vídeo	139
11.2.3.8 Controles de vídeo.....	140
11.2.3.9 Subtítulos activos.....	140
11.2.3.10 Autobúsqueda en el diccionario	140
11.2.3.11 Búsqueda manual en el diccionario.....	141
11.2.3.12 Inserción de sugerencias en el subtítulo.....	141
11.2.3.13 Visualización extendida de sugerencias.....	141
11.2.3.14 Modificación del diccionario.....	142
11.2.3.15 Extracción por lotes de términos y expresiones.....	142
11.2.3.16 Estadísticas de los subtítulos actuales	142
11.2.4 Otras utilidades.....	143
11.2.4.1 Cambio de idioma de la interfaz gráfica de usuario	143
11.2.4.2 Configuración.....	143

Índice de ilustraciones

Ilustración 1: Vocabulario básico español-inglés.....	15
Ilustración 2: Pantalla de MacCaption.....	26
Ilustración 3: Pantalla de EZTitles.....	27
Ilustración 4: Pantalla de PoliScript 3000.....	28
Ilustración 5: Pantalla de WinCaps	29
Ilustración 6: Pantalla de Subbits.....	30
Ilustración 7: pantalla de Subtitle Translation Wizard.....	30
Ilustración 8: Pantalla de Subtitle Workshop	31
Ilustración 9: Pantalla de dotSUB	32
Ilustración 10: Ejemplo de documento XML	36
Ilustración 11: Ejemplo de grafo.....	38
Ilustración 12: Metodología de desarrollo de ontologías	48
Ilustración 13: Cuadro de tareas del diagrama de Gantt	55
Ilustración 14: Diagrama de Gantt	56
Ilustración 15: Diagrama de casos de uso del escenario 1	72
Ilustración 16: Diagrama de casos de uso del escenario 2	74
Ilustración 17: Diagrama de casos de uso del escenario 3	77
Ilustración 18: Diagrama de arquitectura	82
Ilustración 19: Pantalla contenedora MDI	84
Ilustración 20: Pantalla de consulta del diccionario	84
Ilustración 21: Pantalla de edición del diccionario	85
Ilustración 22: Pantalla de traducción de subtítulos	86
Ilustración 23: Pantalla de traducción de subtítulos con vídeo	86
Ilustración 24: Diagrama de clases del componente Presentación	88
Ilustración 25: Diagrama de clases – Negocios: subtítulos	91
Ilustración 26: Diagrama de clases - Negocios: entradas, validación y BOBase	92
Ilustración 27: Diagrama de clases de del componente Negocio	93
Ilustración 28: Diagrama de clases del componente Servicio	94
Ilustración 29: Diagrama de clases del componente Acceso a datos.....	95
Ilustración 30: Diagrama de paquetes del sistema.....	96
Ilustración 31: Relación N-aria no posible en RDF.....	98
Ilustración 32: Relación N-aria en RDF con nodo de relación	98
Ilustración 33: Ontología RDF creada	99
Ilustración 34: Diagrama Entidad-Relación del sistema	101
Ilustración 35: Pantalla principal	131



Ilustración 36: Pantalla de edición de diccionario.....	132
Ilustración 37: Ventana de consulta del diccionario.....	134
Ilustración 38: Ventana de subtitulación – sin subtítulos	136
Ilustración 39: Subtítulo	137
Ilustración 40: Subtítulo con indicación de longitud excesiva	138
Ilustración 41: Marca y anotación en subtítulo	138
Ilustración 42: Ventana de subtitulación.....	139
Ilustración 43: Estadísticas	143
Ilustración 44: Formulario de cambio de idioma.....	143
Ilustración 45: Formulario de configuración – opciones de traducción	145
Ilustración 46: Formulario de configuración - opciones de diccionario.....	147

Índice de tablas

Tabla 1: Aspectos positivos y negativos de la subtitulación	22
Tabla 2: Lista de expresiones sujeto-predicado-objeto	38
Tabla 3: Ejemplo de expresiones RDF.....	40
Tabla 4: Elementos de una expresión RDF.....	41
Tabla 5: Duración del proyecto por fases.....	57
Tabla 6: Salarios.....	58
Tabla 7: Dedicación de personal por fase de proyecto (en horas)	58
Tabla 8: Costes de personal	58
Tabla 9: Gastos materiales	59
Tabla 10: Presupuesto.....	60
Tabla 11: Elementos de ontologías existentes empleados.....	100
Tabla 12: Guía de referencia - botones.....	132
Tabla 13: controles de vídeo.....	140

1. Introducción

1.1 Motivación del proyecto

99 INT. PUENTE

Kirk, preocupado, se dirige a Scott por el intercomunicador.

KIRK
Impulso factorial, Sr. Scott.

VOZ DE SCOTT POR EL INTERCOMUNICADOR
¿Cómo dice?

KIRK
Que active el factor fuerza,
Sr. Scott.

Se produce una breve pausa.

VOZ DE SCOTT POR EL INTERCOMUNICADOR
¿Perdón?

KIRK
(impacientándose)
El motor de curvatura, Sr.
Scott. ¡El motor de curvatura!

VOZ DE SCOTT POR EL INTERCOMUNICADOR
¡Ah, vale!

--

Lo anterior, como se habrá figurado el lector, no es un diálogo real de ningún episodio ni película de *Star Trek*. Lo ha creado la autora del presente proyecto para ilustrar una circunstancia habitual en el sector de la traducción audiovisual: la falta de coherencia, con la consiguiente confusión para el espectador, que en ocasiones se produce en la traducción (o ausencia de tal) de términos recurrentes entre películas y sus continuaciones, temporadas de la misma serie, capítulos de la misma temporada o incluso dentro de un mismo capítulo. Efectivamente, el siempre congruente “warp drive” en inglés ha recibido, a lo largo de la saga, las dispares traducciones al castellano de “impulso factorial”, “factor fuerza”, “motor de curvatura”... y también “deformación”, “factor espacial”, “hiperespacio” y seguramente unas cuantas más. Esto no se limita solo al doblaje, ya que es fácil hallar diferencias entre este, los subtítulos para cine, los hechos para televisión, para DVD, Blu-ray, difusión por internet, etcétera, de un mismo producto.

Cada vez es más habitual que varios traductores participen en la traducción de distintos capítulos de una misma serie o en distintos extras para DVD o Blu-ray de una misma película, a menudo sin tener contacto estrecho, o incluso contacto alguno, con

los demás. También es frecuente que una misma persona traduzca hoy, por ejemplo, un capítulo de *South Park*, mañana un documental sobre dinosaurios, y a la semana siguiente un largometraje sobre la Segunda Guerra Mundial.

El vertiginoso ritmo de producción de los materiales audiovisuales, sujeto a plazos de entrega cada vez más breves, obliga al traductor a dedicar un tiempo limitado a documentarse sobre un determinado tema con el que pueda no estar familiarizado. Esto no es exclusivo del sector audiovisual, ni mucho menos. La búsqueda de la eficiencia y la inmediatez han propiciado la aparición de herramientas de traducción asistida por ordenador, que facilitan el trabajo del traductor y la colaboración entre grupos de traductores y permiten reducir los tiempos de entrega, pero en gran medida el campo de la traducción audiovisual ha permanecido al margen de dichos avances. Erróneamente se puede considerar que, al no tratarse de una traducción técnica ni científica, los diccionarios o glosarios a los que pueda tener acceso el traductor audiovisual son suficientes para realizar un trabajo satisfactorio. Sin embargo, como se ha expuesto unas líneas más arriba, el material al que debe enfrentarse un traductor audiovisual puede ser extremadamente heterogéneo y el tiempo del que dispone, escaso.

Aunque en la actualidad existen gran variedad de diccionarios y glosarios temáticos, al tratarse de herramientas externas, independientes y rígidas, el apoyo real que suponen es muy inferior al que podría obtenerse de un componente único, flexible, ampliable e integrado en la propia aplicación de traducción. Por otra parte, las soluciones de traducción asistida con procesamiento de texto integrado que existen en la actualidad (Trados (1), WordFast (2), OmegaT (3)...), que entre otras cosas proporcionan memorias de traducción y glosarios personalizados y ampliables, son inadecuadas para el uso con material audiovisual, ya que están exclusivamente orientadas a la traducción de documentos.

Es por ello que resulta necesario aunar lo mejor de ambos mundos y proporcionar al sector audiovisual una herramienta que lleve un paso más allá el proceso de traducción, participando al fin de las ventajas de la traducción asistida, pero adaptándolas a sus necesidades particulares.

La autora de este proyecto lleva 10 años ejerciendo como traductora audiovisual, por lo que confía en que su experiencia directa la sitúe en una posición privilegiada a la hora de diseñar la solución.

1.2 Objetivos

Dado que la adaptación de un guión audiovisual para el doblaje a otros idiomas requiere de un proceso complejo, que incluye el ajuste de los diálogos; que además puede realizarse la traducción en un documento compatible con las herramientas de traducción asistida existentes, y que el uso de los subtítulos está más extendido que el doblaje en la mayoría de países, se ha decidido que el objetivo principal del proyecto sea diseñar y desarrollar una aplicación de traducción asistida de subtítulos para el sector audiovisual. Dicha herramienta debe cumplir las siguientes funciones y condiciones:

1. Permitir, además de las funciones propias de una aplicación de traducción de subtítulos, la consulta y modificación de un repositorio o diccionario de términos y expresiones entre el idioma de partida y el idioma objetivo.
2. Incluir, en el mismo diccionario, sinónimos en el idioma objetivo para su uso durante la revisión de la primera traducción realizada, con el fin de corregir errores y mejorar la calidad.
3. La consulta del diccionario debe poder realizarse automáticamente al accederse a cada subtítulo sin intervención del traductor, para no interrumpir el ritmo de trabajo de éste.
4. Deben poder incluirse ámbitos, contextos o ejemplos en cada binomio expresión A - expresión B, para facilitar consultas futuras y proporcionar resultados completos y ajustados al contexto.
5. El diccionario debe plantearse de tal forma que permita, en una segunda fase, el acceso a un mismo diccionario de varios miembros de un mismo equipo o comunidad de traductores.
6. El modelado de las expresiones se realizará mediante un vocabulario estructurado según el Resource Description Framework (RDF) (4), con tripletas sujeto-predicado-objeto. RDF permite representar el conocimiento en una red de nodos, que en nuestro caso serán las palabras o expresiones a almacenar (sujeto y objeto), interconectados por flechas dirigidas, que corresponderán a la relación (predicado) que une esas dos expresiones. Aunque se desarrollará un vocabulario más complejo, un ejemplo básico que ilustraría esta forma de representación entre términos en español y en inglés podría ser:

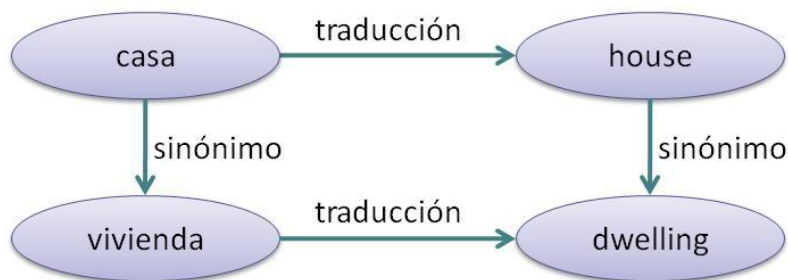


Ilustración 1: Vocabulario básico español-inglés

7. Por último, el programa debe ser fácilmente ampliable, por lo que se empleará una arquitectura modular y por capas que permita futuras modificaciones y mejoras.

1.3 Estructura del documento

A continuación se expone la estructura que adoptará el presente documento, a modo de orientación para el lector. Los capítulos de los que constará esta memoria son:

- **1. Introducción:** se realiza una sucinta exposición del problema que trata de resolverse y de los objetivos que tratará de cumplir el proyecto.
- **2. Estado de la cuestión:** para comprender la esfera de aplicación de la herramienta que se pretende desarrollar, se describen las tecnologías de traducción asistida existentes en la actualidad. Asimismo, se presentan las peculiaridades de la traducción audiovisual, y se hace una revisión de los sistemas de edición y traducción de subtítulos.
- **3. Representación del conocimiento y ontologías:** en este capítulo se explica brevemente en qué consiste y qué función tiene la representación del conocimiento, y se revisan los conceptos más relevantes y algunos de los sistemas de representación y razonamiento existentes..
- **4. Planificación del proyecto:** en el cuarto apartado se diseña un plan de trabajo, especificando las tareas de las que constará el proyecto. También se realizará un presupuesto detallado.
- **5. Análisis del problema:** se analizará la problemática actual y se especificarán los requisitos que debe cumplir el sistema.
- **6. Diseño de la solución:** se realizará un diseño completo del sistema. Se describirá el modelo empleado para representar la información en el repositorio.



- **7. Implementación:** se detallarán en este capítulo los elementos más significativos de la implementación del sistema, y se comentarán los aspectos de mayor relevancia.
- **8. Evaluación de la herramienta:** se evaluarán las ventajas y desventajas que supone el uso de la herramienta para el traductor de subtítulos.
- **9. Trabajos futuros y conclusiones:** aquí se expondrán las vías de desarrollo posterior que podrían seguirse para la ampliación y mejora de la herramienta, así como las conclusiones alcanzadas tras la elaboración del proyecto.
- **10. Bibliografía:** se refieren en este apartado las fuentes de documentación empleadas para la elaboración del proyecto.
- **11. Apéndices:** se incluyen algunas consultas efectuadas al repositorio desde la aplicación, así como el manual de usuario de la aplicación.

2. Estado de la cuestión

2.1 Traducción asistida por ordenador

2.1.1 Traducción automática y traducción asistida por ordenador

En las últimas décadas se han producido cambios sustanciales y vertiginosos en prácticamente todos los ámbitos de la vida humana, especialmente en las sociedades más modernas. Los avances tecnológicos han permitido una globalización sin precedentes, de tal suerte que lugares y culturas geográficamente remotos se encuentran hoy a unas pocas horas de vuelo, o incluso a un simple clic del ratón.

Este acercamiento supone el desarrollo de un mercado global a una escala nunca vista, y las diferencias lingüísticas no pueden ser una barrera para el flujo de bienes, servicios y personas. Si bien la lengua inglesa ha adquirido una gran presencia a nivel mundial, es importante preservar la diversidad lingüística y cultural de cada región del planeta. Es por ello que, frente al incremento de los intercambios internacionales, el papel de las traducciones ha cobrado, si no más importancia, desde luego más ubicuidad a todos los niveles.

En este contexto, no es de extrañar que se pretenda que ese mismo desarrollo tecnológico juegue un papel fundamental en el proceso de traducción, para mejorar la productividad y la calidad y a la vez reducir los costes. Este fenómeno no es tan reciente como cabría esperar: hace ya más de 50 años que la posibilidad de delegar la traducción en las máquinas suscitó un gran interés entre organismos e instituciones diversas. Desde entonces se han llevado a cabo investigaciones de diversa índole, orientadas a mejorar la calidad de las traducciones creadas sin intervención humana. Comenzando en sus inicios con el empleo de extensos diccionarios bilingües y una serie de reglas para ordenar las palabras, pasando por los modelos sintácticos o el cambio de estrategia de los años 90 hacia el uso de textos y ejemplos, el campo de la traducción automática ha registrado importantes avances, pero a pesar de todos los esfuerzos, continúa representando un porcentaje marginal en el mercado de la traducción. Únicamente cuando se aplica a un contexto muy específico y restringido, con un vocabulario limitado y estructuras gramaticales repetitivas, se consigue una traducción suficientemente satisfactoria. Un buen ejemplo de ello sería el sistema de traducción del pronóstico meteorológico MÉTEO (5), en existencia desde los años 80. En la mayoría de casos, pues, la intervención humana sigue siendo necesaria, bien revisando y corrigiendo una traducción automática, bien realizándola completamente.

El desencanto con la traducción automática ha llevado a un nuevo planteamiento del puesto que debe ocupar la tecnología, salvo casos puntuales, en el campo de la traducción: el de servicio al traductor, para auxiliarle en su labor y mejorar su productividad. Esto es lo que se conoce como traducción asistida por ordenador o Computer-Aided Translation (CAT) (6).

En su definición más amplia, la traducción asistida por ordenador comprendería cualquier herramienta computarizada de la que pueda hacer uso un traductor para realizar su trabajo, entre ellas los buscadores de Internet, los procesadores de texto o el reconocimiento de voz (6). Sin embargo, dado que muchas de estas herramientas son ya de uso generalizado entre buena parte de la población, y en cualquier caso no son específicas para los profesionales del lenguaje, se dejarán fuera de esta breve introducción.

Puesto que no se pretende hacer un estudio exhaustivo de todas las herramientas disponibles, se repasarán únicamente las que gozan de mayor aceptación entre los traductores, y que han cobrado más auge en los últimos años: los diccionarios y sistemas de gestión de bases de datos terminológicas (6), las herramientas de análisis de corpus (7) y las memorias de traducción (8).

2.1.2 Diccionarios electrónicos y sistemas de gestión de bases de datos terminológicas

El diccionario ha sido una herramienta fundamental para el traductor desde los comienzos de la profesión. Sin embargo, la aparición de los diccionarios electrónicos, ya sea en formato CD-Rom, aplicaciones instalables o disponibles por Internet, ha supuesto una auténtica revolución en el ejercicio de la profesión. Las ventajas son múltiples: la consulta de términos es mucho más rápida que en papel; se facilita la búsqueda de palabras a partir de sus lemas; se puede acceder a información relacionada a través de enlaces... Además, se pueden consultar a través de Internet diccionarios, glosarios y bases de datos terminológicas en abundancia, tanto bilingües como monolingües, que proporcionan al traductor acceso inmediato a una variedad nunca vista de terminología especializada desde un ordenador personal. Una base de datos de especial relevancia sería IATE (InterActive Terminology for Europe) (9), que en la actualidad contiene aproximadamente 1,4 millones de entradas multilingües de una amplísima diversidad de ámbitos especializados. Su existencia permite una mayor estandarización de la terminología empleada por las instituciones de la Unión Europea y, por extensión, por cualquier traductor que haga uso de ella.

Para hacer un mejor aprovechamiento de los diccionarios mencionados, los sistemas de gestión terminológica o TMS (Terminology Management System) permiten almacenar en un repositorio términos que vaya recopilando el traductor durante su trabajo, y consultarlos posteriormente. Si bien requieren una inversión de tiempo para construir la base de datos, a la larga suponen un ahorro considerable, y además proporcionan una mayor coherencia y calidad lingüística a la traducción.

Algunos de estos sistemas, sobre todo los que están integrados en una solución completa de traducción (es decir, con procesador de texto y memoria de traducción), ofrecen una utilidad muy conveniente al usuario: el llamado "reconocimiento automático de terminología", que consiste en la búsqueda automática del contenido del texto en el repositorio a medida que el traductor va avanzando con su traducción. De esta forma, se ofrecen las coincidencias al usuario, que puede emplearlas si así lo desea. Otras herramientas llegan incluso a procesar por lotes el texto completo y a generar una traducción preliminar de este, que el traductor podrá editar y completar posteriormente.

Una última utilidad disponible en varias de estas herramientas es la extracción automática de términos. La aplicación analiza el texto completo en busca de bloques de palabras y facilita al traductor una lista para su edición y la posible inclusión de los términos en la base de datos, lo que supone un ahorro de tiempo importante. Aunque la mayoría de estos componentes realizan un análisis monolingüe, existen ya herramientas que cotejan el texto traducido con el original en busca de posibles correspondencias.

2.1.3 Herramientas de análisis de corpus

En el contexto que nos ocupa, los corpus (7) son grandes colecciones de textos electrónicos compilados de acuerdo a unos criterios concretos para asistir al traductor, que no tiene por qué dominar un tema determinado, en cuanto al formato, estilo, terminología y fraseología empleados habitualmente por los expertos en ese campo.

Las herramientas de análisis de corpus proporcionan diversas funciones al traductor para extraer información relevante del corpus de textos recopilados. A continuación se hará una exposición somera de las más utilizadas: las listas de frecuencia de palabras y las concordancias.

Las listas de frecuencia de palabras (6) proporcionan al traductor una lista con todas las palabras existentes en el corpus, y la cantidad de apariciones de cada una.

De esta forma, por ejemplo, si el traductor dispone de varias traducciones posibles para un término y no está seguro de cuál se emplea habitualmente, puede comprobar la frecuencia de aparición de cada una de sus opciones y así elegir la más adecuada. Algunas herramientas más avanzadas calculan también la longitud media de palabras, frases y párrafos. Con estos datos, el traductor puede hacerse una idea de los rasgos estilísticos de los textos analizados, y tenerlo en cuenta a la hora de realizar su traducción.

Los buscadores de concordancias (6) hallan todas las incidencias de un patrón de búsqueda y lo muestran centrado en pantalla, junto con parte del texto que lo acompaña. Esto puede resultar útil para ubicar palabras en su contexto y comprobar, por ejemplo, cuáles son los adjetivos que más se emplean junto a determinado nombre.

2.1.4 Memorias de traducción

A diferencia de las herramientas de análisis de corpus, las de gestión de memorias de traducción suelen estar integradas en soluciones completas de traducción, como Trados (1) u OmegaT (3). Su función es crear, a partir de traducciones ya realizadas, bases de datos con segmentos (habitualmente frases) en el idioma original y en el idioma destinatario, de manera que puedan ser reutilizados por el traductor. De esta forma, cuando se halla un segmento idéntico o similar a alguno almacenado, la herramienta propone automáticamente una correspondencia, que puede insertarse a voluntad del usuario.

Por su naturaleza, las memorias de traducción se prestan más al uso con documentos con un contenido elevado de frases repetidas. Tal es el caso de manuales de usuario o versiones actualizadas de páginas web ya existentes. Además, si se hace un uso indebido de la herramienta y no se revisa adecuadamente la traducción, el texto podría pecar de falta de cohesión, y acabar siendo poco más que una mezcolanza de frases sin nexo de unión alguno (8).

Es preciso señalar, pues, que las memorias de traducción integradas en la aplicación, así como el resto de herramientas de traducción asistida, se deben utilizar de forma responsable y hoy por hoy no deben pasar de ser una ayuda, un valioso accesorio, a la labor principal, que es la del traductor. Si el peso de la traducción acaba recayendo sobre estos complementos, por reducir costes o acortar excesivamente los plazos de entrega, se corre el riesgo de sufrir una merma

importante en la calidad, incurriendo precisamente en aquello que se pretendía evitar mediante el uso de estas herramientas.

2.2 Traducción audiovisual

2.2.1 Aspectos generales

El término “traducción audiovisual” (TAV) se refiere a la modalidad de trasvase lingüístico en la que se conjugan, además del código lingüístico, los códigos visual y oral (10). Esta peculiaridad inherente la diferencia de otras modalidades traductorales más tradicionales, ya que impone ciertos condicionantes a la hora de llevar a cabo el proceso de traducción. Así, los condicionantes espacio-temporales, la sincronía que forzosamente ha de haber con imagen y sonido, así como la exigencia de menor literalidad en favor de la fidelidad narrativa, o la imposibilidad de recurrir a elementos explicativos, tales como las notas a pie de página, componen un marco de trabajo algo particular.

Tampoco es baladí la dificultad a la hora de especializarse en este campo. La formación específica puede proveer al traductor de conocimientos técnicos y prácticos para ejercer su labor adecuadamente, pero por fuerza la especialización terminológica es imposible de alcanzar en esta modalidad, que abarca temáticas extraordinariamente diversas.

No obstante, se trata de un sector con unas perspectivas de futuro halagüeñas, a juzgar por el constante aumento en la oferta y demanda de contenidos audiovisuales. Al vivir en una sociedad donde cada vez más se busca estimular los sentidos para llegar al receptor, es normal que gran parte de la información que antes se transmitía por medio escrito aproveche ahora las posibilidades de la tecnología multimedia. En concreto, la Unión Europea, uno de los mayores consumidores de traducción del mundo, está investigando la posibilidad de diversificar su material informativo y poner más énfasis y recursos en la producción e investigación de material audiovisual (10). Si a ello se une la diferenciación de los canales televisivos y la multiplicación de los formatos de difusión, se podrá afirmar que la traducción audiovisual no está abocada a desaparecer en una temporada.

Cabe aquí subdividir la práctica de la traducción audiovisual en diversas modalidades, de las cuales tres gozan de una mayor aplicación a un nivel internacional: el doblaje, el *voice-over* y la subtitulación. Tradicionalmente, en España (así como en otros grandes países europeos como Alemania o Francia) se ha favorecido el doblaje en detrimento de las otras opciones, pero en los últimos años ha

ido cobrando fuerza la distribución de productos subtítulados. Esto, sumado a la clara preferencia por la subtitulación de países más pequeños como Holanda, Portugal o los países escandinavos, y a la presión temporal que oprime al sector, justifica la creación de una herramienta que facilite la labor de los traductores de esta modalidad.

2.2.2 Subtitulación

La subtitulación y el doblaje (las dos técnicas más empleadas) ofrecen distintas ventajas e inconvenientes que pueden motivar la predilección por una u otra. En la siguiente tabla se ofrece un compendio de los aspectos positivos y negativos del subtítulo (10):

SUBTITULACIÓN	
ASPECTOS NEGATIVOS	ASPECTOS POSITIVOS
<ul style="list-style-type: none"> • Contamina la imagen original • Traslada el código oral cinematográfico a un formato escrito no propio del género • Dispersa la atención del espectador • Es inadecuado en regiones con un índice de analfabetismo elevado • Se condensa el texto original • No permite el solapamiento de diálogos • Subordinado a limitaciones espacio-temporales • Se mantienen dos lenguas distintas a la vez, una oral y otra escrita, lo que puede producir confusión 	<ul style="list-style-type: none"> • Bajo coste • Contribuye al aprendizaje de idiomas • Proceso más rápido que el doblaje • Mantiene las voces originales de los actores, por lo que se disfruta de su interpretación íntegra • Mejor para personas con problemas de audición y para inmigrantes • Se puede utilizar con cualquier programa audiovisual

Tabla 1: Aspectos positivos y negativos de la subtitulación

Aunque uno de los factores que hacen decantarse a muchas productoras y distribuidoras por esta opción es el puramente económico, sus ventajas para el espectador, a pesar de los aspectos negativos, también contribuyen a que la demanda de estos productos vaya gradualmente en aumento.

Este incremento en su uso ha de llevar aparejada, si se desea una buena acogida entre el público, una calidad, tanto técnica como lingüística, a la altura del producto. Las reducciones presupuestarias y las presiones temporales no pueden llegar al extremo de menoscabar la calidad del resultado final, pero también ha de ser bienvenida toda aportación que contribuya a agilizar el proceso, siempre que se garantice un nivel de calidad adecuado.

2.2.3 Práctica profesional

2.2.3.1 *Proceso de subtitulación*

El proceso de subtitulación, desde que se produce el encargo hasta que se transmite por el medio correspondiente, es necesariamente un proceso multidisciplinar, donde diversos profesionales contribuyen a la creación del subtítulo en cada una de sus fases.

Se centrará este apartado en la figura del traductor, en su rol tradicional, así como en otras funciones que en ocasiones se solapan con la función de traducción propiamente dicha. Las fases en las que puede intervenir el traductor son las siguientes:

- **Localización o pautado:** consiste en el proceso de asignar códigos de tiempo de entrada y salida a los diálogos del programa, pautando así las unidades traducibles. Para ello es necesario disponer del vídeo con código de tiempos y de una lista de diálogos. En caso de no disponer de ella, o de que ésta sea obsoleta por haberse elaborado antes del montaje final, el localizador habrá de corregir las discrepancias o incluso transcribir todos los diálogos.
- **Traducción:** si el traductor dispone de un programa de subtitulación que le permite visionar el texto sobre la imagen, la traducción podrá ser mucho más ágil y efectiva. En caso contrario, deberá realizar una cuidadosa toma de notas durante el primer visionado, apuntando contextos o aclarando posibles lecturas ambiguas, para así poder realizar la traducción sin recurrir a la imagen. Este paso de toma de notas va quedando obsoleto, a medida que la digitalización del vídeo y los programas de subtitulado, que permiten una sincronización perfecta, van ampliando su cuota de mercado.
- **Adaptación:** tarea que consiste en condensar los subtítulos para ajustarlos a las limitaciones espacio-temporales correspondientes. Estas dependerán del tiempo que se mantenga en escena el subtítulo y la velocidad de lectura que se le presupone al espectador medio.
- **Revisión:** puesto que el traductor es un ser humano falible, conviene revisar todo el texto para verificar que no hay faltas ni errores y para mejorar la traducción allí donde sea posible.

El proceso de subtitulación no es en absoluto uniforme, de modo que dependiendo del estudio de subtitulación del que se trate, algunos de los pasos anteriores pueden omitirse, modificarse o combinarse. Igualmente, puede darse tanto

el caso de que una persona distinta se encargue de cada fase, como la situación de que sea la misma persona, el subtitulador, la que lleve a cabo todos y cada uno de los pasos descritos.

Puede considerarse que la localización y la traducción deben recaer en el mismo profesional. Así, tendrá libertad absoluta a la hora de dividir los diálogos en subtítulos de la forma que resulte más natural en la lengua meta. Sin embargo, unir estas dos figuras puede presentar también varias desventajas: la primera de ellas, económica, puesto que resulta más rentable realizar un único pautado en el idioma original, lo que se conoce como *master titles* (10), y distribuirlo a los estudios de subtitulación nacionales (salvo que se ocupe un solo estudio internacional) que realizar una nueva localización para cada lengua meta. Igualmente, ello permite, en caso de no contar con una lista de diálogos, que realice la localización un nativo que pueda entender con más facilidad las voces distorsionadas, los acentos regionales cerrados o los diálogos que se solapan con más facilidad y sea menos propenso a cometer errores de comprensión que un traductor.

En este documento no se aboga, pues, por incluir la localización entre las tareas del traductor/subtitulador, sino que se considera que sería más práctico y eficaz permitirle modificar la localización de los subtítulos puntuales que considere conveniente alterar. En cuanto al resto de tareas, sí que se defiende aquí la fusión de traductor, adaptador y revisor en una misma figura. Resulta evidente que el adaptador, encargado de condensar el texto, solo podrá hacer una buena labor si domina el idioma de partida tanto como el traductor, para evitar la introducción accidental de omisiones graves o errores. Asimismo, si bien conviene que una segunda persona colabore en la revisión, el traductor siempre tendrá que revisar su trabajo antes de entregárselo al cliente o estudio.

2.2.3.2 *Limitaciones espacio-temporales*

A continuación se enumerarán una serie de consideraciones técnicas (10) que debe tener en cuenta el subtitulador, y por ende, la autora de este proyecto a la hora de diseñar su solución:

- **Número de líneas:** para no obstruir en exceso la imagen, se ha generalizado el máximo de dos líneas por subtítulo.
- **Presentación y posicionamiento de los subtítulos:** en este caso, el posicionamiento centrado o justificado a la izquierda de los subtítulos, así como la ubicación de los subtítulos monolineales en la línea superior o en la

inferior, queda básicamente en una cuestión de gustos, siempre que se mantenga la coherencia a lo largo de todo el programa.

- **Número de caracteres por línea:** en función del canal de distribución y de los criterios del cliente, el número máximo de espacios y caracteres por línea debe mantenerse entre 28 y 40, para permitir un tamaño de letra adecuado.
- **Tipo de letra:** por cuestiones de legibilidad, en general se trabaja con un tipo de letra sin gracia, como Helvetica, Arial y Times New Roman.
- **Subtítulos cortos:** para evitar la aparición fugaz del subtítulo en pantalla, lo recomendable es que se mantenga durante al menos un segundo. Esto también condiciona el número mínimo de caracteres, que de forma ideal no debe ser inferior a 4 ó 5 caracteres. Un número menor llevará al espectador a releer el texto involuntariamente.
- **Tiempo máximo de permanencia en pantalla:** teniendo en cuenta la velocidad de lectura del público, se recomienda que un subtítulo monolineal se mantenga en pantalla no más de 4 segundos, y uno bilineal no más de seis segundos. Una duración mayor provocará, como en el caso anterior, la relectura del subtítulo por parte del espectador.
- **Relación segundos/caracteres:** no existe una regla inmutable que determine la duración que deben los subtítulos, ya que influyen muchos factores en la velocidad de lectura que se le presupone a la audiencia: el público al que va destinado el producto (niños, personas mayores, con distinto nivel cultural, etc.), el canal de distribución (en una sala de cine hay menos distracciones que en casa, ante el televisor) o la familiaridad que tenga la audiencia con la subtitulación. Dicho esto, sí que existe una regla que se utiliza habitualmente como directriz: la regla de los seis segundos. Según ella, en un período de seis segundos el espectador es capaz de leer y asimilar la información contenida en dos líneas de subtítulos, cuando cada línea contiene un máximo de 35 espacios o pulsaciones. Una sencilla regla de tres bastará para calcular el número máximo de pulsaciones para la duración de cada subtítulo determinado.

2.3 Herramientas existentes para la traducción de subtítulos

Al tener un grupo de destinatarios tan específico, el desarrollo de este tipo de herramientas se concentra, salvo contadas excepciones, en un conjunto de empresas y estudios que realizan servicios profesionales para el sector audiovisual. Algunas de las aplicaciones creadas son, incluso, de uso exclusivamente interno. Al tener un

enfoque empresarial, los precios de las aplicaciones destinadas a su comercialización son, en general, bastante elevados, del orden de los 1.500 - 2.000 euros (11).

También existen en el mercado algunas herramientas de corte menos profesional que, pese a ofrecer una funcionalidad más limitada, están destinadas a la edición o traducción de subtítulos. En las siguientes líneas se realizará una exposición de algunas de estas herramientas de ambas categorías y de sus características principales.

2.3.1 MacCaption



Ilustración 2: Pantalla de MacCaption

MacCaption (12) es una herramienta para la importación, creación, edición y exportación de subtítulos. Algunas de sus características son:

- Plataforma: MacOS.
- Soporta múltiples formatos.
- Permite la visualización sincronizada.
- Conversión de códigos de tiempo.
- Modificación de códigos de tiempo.
- Ajuste de la velocidad de lectura.
- Detección de escenas.
- Corrector ortográfico.

- En las versiones más completas del programa, se ofrece procesamiento por lotes, interfaz de línea de comandos, subtítulo remoto o impresión en cinta de vídeo vía Firewire.

2.3.2 EZTitles

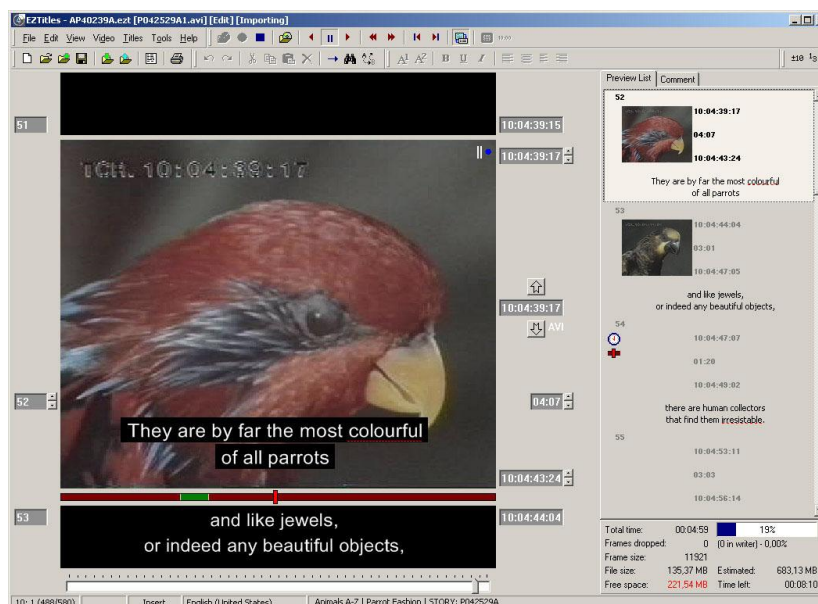


Ilustración 3: Pantalla de EZTitles

EZTitles (13) es, como la anterior, una solución completa de subtitulación. Sus características principales son:

- Plataforma: Windows XP/Vista/7.
- Soporta múltiples formatos.
- Permite crear un formato personalizado.
- Permite operación exclusiva desde teclado, mediante combinaciones de teclas.
- Corrector ortográfico (requiere Microsoft Word u Open Office).
- Permite buscar y reemplazar porciones de texto.
- Permite captura de vídeo.

2.3.3 PoliScript 3000

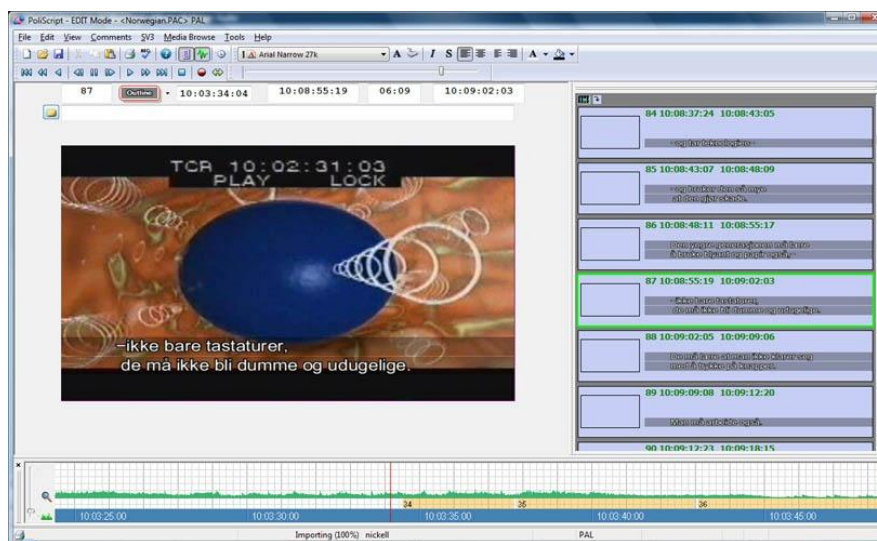


Ilustración 4: Pantalla de PoliScript 3000

PoliScript 3000 (14) es un programa integral de preparación de subtítulos, que cuenta con las siguientes características:

- Plataforma: Windows XP y Vista.
- Soporta múltiples formatos.
- Corrector ortográfico.
- Generación de códigos de tiempo.
- Se puede alquilar por horas, para proyectos puntuales.
- Generación de subtítulos para DVD.
- Indicadores de velocidad de lectura.
- La versión más completa del programa incluye: reproducción a alta velocidad, función de “ir a...” número de subtítulo y código de tiempo, pistas de audio alternativas o captura de vídeo, entre otras funciones avanzadas.

2.3.4 WinCaps

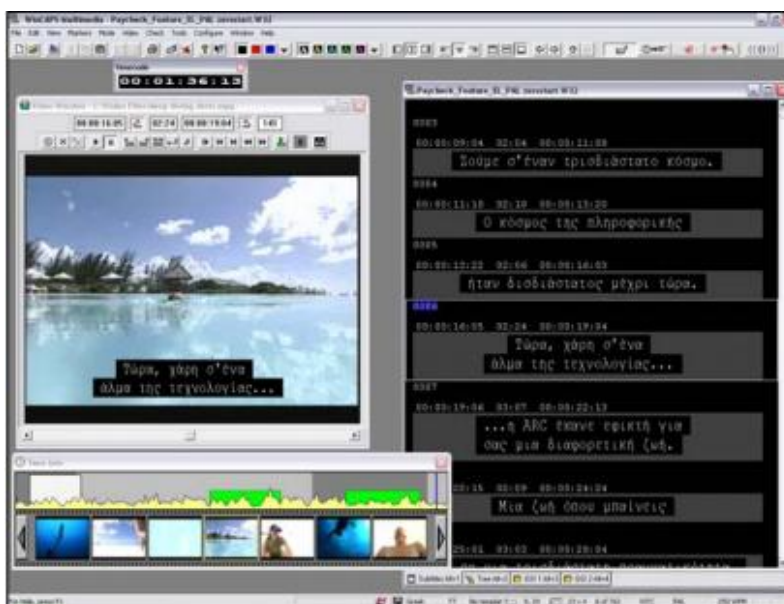


Ilustración 5: Pantalla de WinCaps

WinCaps (15) es, al igual que las soluciones anteriores, una herramienta completa de subtítulos. Sus características son:

- Plataforma: no especificada.
- La opción AutoTime alinea el texto del subtítulo con la pista de audio.
- Múltiples formatos.
- Detección de cambio de plano.
- Reconocimiento de voz.
- Posibilidad de personalizar la distribución de las ventanas y las barras de herramientas.
- Permite la inclusión de anotaciones o comentarios en cada subtítulo.

2.3.5 Herramientas semiprofesionales

Se han dejado fuera de este repaso otras soluciones similares a las anteriores, como TV2003 (11), Swift Create (16) o Subeasy (17), por considerarse que resultaría redundante su estudio al ir en la misma línea que las ya revisadas.

Sin embargo, sí que existen en el mercado otras herramientas bastante más asequibles, si bien, en todos los casos, tanto su funcionalidad como su compatibilidad son notablemente más limitadas. Puesto que en ocasiones sí que son utilizadas por traductores profesionales que no tienen acceso a soluciones más sofisticadas, se incluyen algunas de ellas en esta comparativa.

2.3.5.1 Subbits

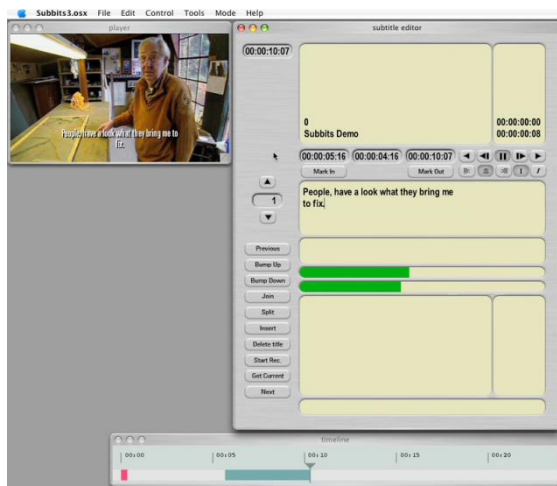


Ilustración 6: Pantalla de Subbits

Subbits (18) es una sencilla aplicación para crear y editar subtítulos. Como las anteriores, se centra más en la localización, ya que permite ir pautando cada subtítulo a partir de la imagen. Las características principales son:

- Plataforma: MacOS X y Windows
- Varios formatos de subtítulos, aunque abanico bastante más reducido que los programas anteriores.
- Incorpora un sistema de información al usuario sobre la relación entre la longitud del texto del subtítulo actual y la longitud ideal. Alerta al traductor cuando excede el 10% y el 20% de la longitud ideal.

2.3.5.2 Subtitle Translation Wizard

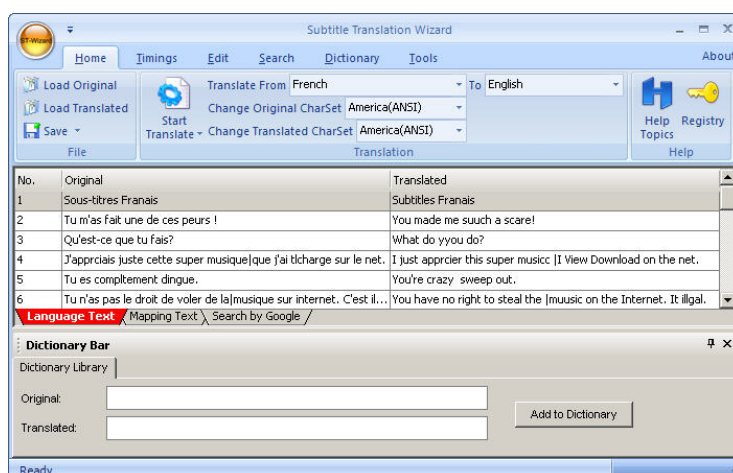


Ilustración 7: Pantalla de Subtitle Translation Wizard

Subtitle Translation Wizard (19) es una sencilla aplicación para la traducción automática de subtítulos. Sus características principales son:

- Plataforma: Windows
- Pocos formatos de subtítulos. No soporta los estándares empleados en la industria.
- Gran variedad de idiomas de partida y de llegada.
- Visualización simultánea de los subtítulos origen y destino.
- Realiza la traducción automática del archivo de origen por medio de Google Translate, y permite la edición del archivo generado.
- No permite reproducción de vídeo.
- No ofrece la posibilidad de limitar la longitud del subtítulo, de incluir marcas o anotaciones durante el trabajo ni de modificar el formato de la letra.
- Incluye un diccionario básico en el que el usuario puede introducir las expresiones que desee en el idioma original, con su correspondencia en el idioma destino. Sin embargo, se contempla únicamente como una herramienta de aprendizaje del idioma, puesto que no se permite la consulta directa en el diccionario.

2.3.5.3 Subtitle Workshop



Ilustración 8: Pantalla de Subtitle Workshop

Subtitle Workshop (20) es una herramienta de edición de subtítulos en formato texto. Permite generar un archivo vacío a partir del archivo fuente para proceder a su traducción.

- Plataforma: Windows
- Permite reproducir el vídeo y superponer los subtítulos originales o los traducidos.

- Permite el formateado del estilo del texto.
- Visualización simultánea de subtítulos origen y destino.
- Amplia compatibilidad con diversos formatos. Basta con actualizar una librería .dll para mantenerlo al día.
- El más completo en cuanto a opciones de edición de texto y códigos de tiempo.
- No ofrece la posibilidad de limitar la longitud del subtítulo, de incluir marcas o anotaciones durante el trabajo ni de modificar el formato de la letra.

2.3.5.4 dotSUB

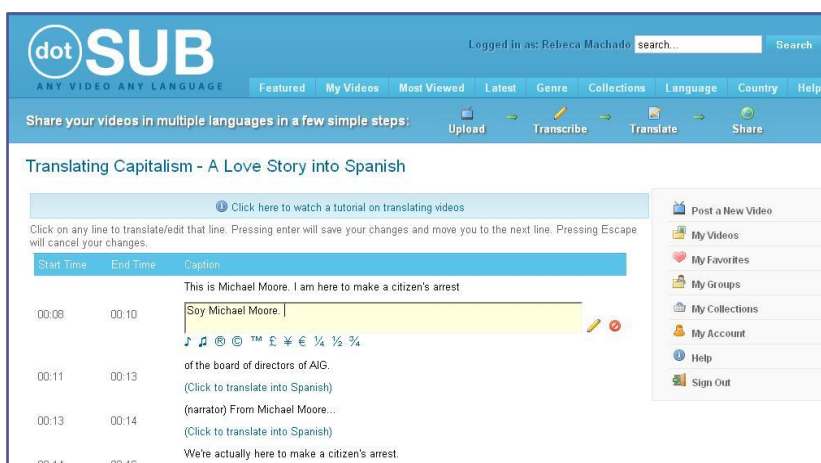


Ilustración 9: Pantalla de dotSUB

Solución en línea (21) para crear y visualizar subtítulos desde un navegador. Consta de las siguientes características:

- Es gratuito para usuarios particulares; ofrecen una solución de gestión de proyectos para empresas, además de servicios profesionales completos de transcripción, subtitulación y traducción.
- Plataforma: todas. Todo el trabajo se realiza desde un navegador.
- Simplicidad. Todo el proceso consiste en cuatro pasos sencillos: subir el material, transcribir, traducir y compartir.
- Permite la colaboración entre varios traductores para realizar una misma traducción, que se almacena en sus servidores durante todo el proceso.
- No permite visualizar el vídeo mientras se está traduciendo, por lo que se dispone de un contexto reducido al realizar la traducción.
- No permite formatear el texto (cursiva, negrita...).
- Guarda el trabajo automáticamente.

2.3.6 Comparativa entre las herramientas existentes y los objetivos planteados

Tras estudiar las aplicaciones anteriores, resulta evidente que ninguna cumple el principal objetivo de este proyecto: proporcionar una utilidad de gestión terminológica integrada en la herramienta de traducción para asistir al traductor durante todo el proceso y mejorar su productividad.

La mayoría de ellas, sobre todo las herramientas profesionales, tampoco permiten visualizar simultáneamente los subtítulos en la lengua de partida y la traducción, por lo que las revisiones exigen una visualización del vídeo para confirmar que la traducción realizada es correcta. Se centran, tal como se ha expuesto, más en el proceso de creación de subtítulos que en el de traducción propiamente dicha.

Si bien la posibilidad de crear el archivo de subtítulos a partir del vídeo (y a veces del guión no pautado) resulta muy útil cuando no se dispone del archivo de subtítulos en el idioma original, ya existe una amplia variedad de aplicaciones que permiten realizar dicha labor. Actualmente, en plena revolución digital y de las tecnologías de las comunicaciones, los distintos elementos que participan en el proceso audiovisual tienen más facilidades que nunca para intercambiar información. Es por ello que cada vez más empresas siguen la práctica de crear un archivo de subtítulos matriz en el idioma original, en un formato de fácil transmisión e intercambio con los diversos departamentos, estudios, colaboradores o clientes. De esta forma se reduce el coste y el tiempo necesario para la creación de los subtítulos, puesto que basta con pautar los subtítulos una sola vez antes de proceder a la traducción a los idiomas que se desee.

En cuanto a la traducción automática que se puede obtener con una de las aplicaciones analizadas, conviene resaltar que, si bien la traducción automática tiene su utilidad en otros ámbitos más restringidos, en la actualidad supone un obstáculo, más que una ayuda, en el mundo audiovisual. Primero, porque el material original no suele ser repetitivo, por lo que el aprendizaje que requeriría la máquina para realizar una traducción aceptable en el futuro sería largo y poco productivo. Segundo, porque el tipo de lenguaje varía de una obra a otra e incluso dentro de la misma obra. Un buen traductor ha de distinguir en cada momento si se trata de habla culta, coloquial, vulgar, formal, informal, argot, jerga, etcétera, y realizar una traducción acorde con el registro. Tercero, porque por mucho que se revise la traducción creada, siempre habrá mayor probabilidad de que se pase por alto algún error o inexactitud que si la



traducción la hubiera realizado una persona. Y por último, porque si bien en toda traducción se busca la fidelidad al original, en la traducción artística, que es donde se enmarca la audiovisual, la fidelidad en las palabras está siempre supeditada a la fidelidad en las ideas y los sentimientos transmitidos. No es tan importante la literalidad en la traducción lingüística como saber plasmar correctamente la visión del creador original de la obra.

3. Representación del conocimiento

3.1 Introducción

El campo de la representación del conocimiento es sumamente amplio y complejo, tanto que incluso resulta difícil dar algo en teoría tan elemental como una definición clara y unificada. Es éste un ámbito que bebe de disciplinas tan dispares como las matemáticas, la psicología, la estadística, la economía, la lingüística, la biología, la filosofía, la lógica o, cómo no, la informática (22). Cada una de ellas tiene su propia visión del mundo, y canalizarlas todas para obtener una definición consensuada y exhaustiva se antoja excesivamente enrevesado y, en todo caso, innecesario para los propósitos de este documento.

Nos conformaremos, pues, con establecer la representación del conocimiento como la especialidad que se ocupa de la forma en que se almacena o representa la información sobre una situación determinada y sus elementos, de tal manera que un programa pueda manipular dicha información e inferir conclusiones a partir de esta.

Para comprender en qué lugar se encuentra hoy en día la representación del conocimiento en su vertiente más práctica, se repasarán brevemente los sistemas más empleados en la actualidad, tal como se resumen en (23).

La tecnología que goza de mayor ubicuidad son los sistemas gestores de bases de datos relacionales (RDBMS). Estos sistemas permiten almacenar gran cantidad de datos en tablas, habitualmente en filas de longitud fija, y realizar búsquedas de forma rápida y efectiva, gracias al empleo de referencias cruzadas y el cálculo relacional. Aunque su efectividad a la hora de almacenar y recuperar la información es elevada, también se trata de un medio rígido y muy estructurado.

Otra tecnología más reciente que ha cobrado mucho auge desde la llegada de Internet son los lenguajes de etiquetado, como el Lenguaje de Etiquetado Extensible (XML) (24). Se trata de un lenguaje concebido para estructurar la información jerárquicamente (en un árbol ordenado), que se sirve de etiquetas y atributos para marcar el contenido. A diferencia de otros lenguajes de etiquetado como el HTML (25), este no especifica cuáles deben ser esas etiquetas y cuál es la función de cada una, sino que su interpretación corresponde a la aplicación que lea el documento. Un ejemplo sencillo de documento XML sería:

```
<?xml version="1.0" encoding="UTF-8"?>
<memoria
  <título>Herramienta para la traducción asistida
    en la industria audiovisual</título>
  <capítulo>
    <título>Representación del conocimiento</título>
    <sección>
      <título>Introducción</título>
    </sección>
  </capítulo>
</memoria>
```

Ilustración 10: Ejemplo de documento XML

XML posee una serie de ventajas que han contribuido a generalizar su uso. Entre ellas, podría resaltarse su comodidad, ya que basta con un editor de texto para generar o revisar un archivo en este formato; no es necesario acceder a ningún servidor ni a ninguna base de datos para utilizarlo; y al almacenarse en archivos de texto, cualquier persona puede leerlos y descifrar con relativa facilidad su contenido. No obstante, también presenta algunas desventajas, como un espacio de almacenamiento elevado, o una eficiencia bastante menor que la de los RDBMS.

Otra objeción importante que se puede hacer a este lenguaje, puesta de manifiesto en (26) es que, dado que XML no ofrece ninguna regla estándar para la representación semántica de la información, se observa que incluso una afirmación sencilla se puede expresar de muchas maneras. Igualmente, no resulta inmediato, a partir del documento XML, extraer unívocamente la información presentada. Sin conocer exactamente la especificación del vocabulario empleado, el programa que lea el documento no puede saber cómo interpretar la información ni cuáles son las relaciones presentes. Además, como hay muchas maneras de trasladar el árbol lógico a expresiones XML, si se desea hacer una consulta, ésta tendrá que ser muy compleja y contemplar todas las posibilidades de representación.

Estrechamente ligada a la anterior, en la última década ha surgido una tecnología que, sin estar concebida como sustituto de ninguno de los sistemas mencionados, ha ido ocupando un lugar complementario a estos. Se trata de Resource Description Framework (RDF) (4), y aunque su alcance no es ni mucho menos comparable aún a los anteriores, poco a poco va abriéndose camino en el mundo de la representación del conocimiento.

3.2 RDF(S) y ontologías

3.2.1 Resource Description Framework (RDF)

RDF nació con el objetivo de codificar metadatos sobre los documentos accesibles en Internet. Incluir en el documento etiquetas que especificaran datos como el autor, el idioma, la fecha de modificación o el formato facilitaba su consulta y el intercambio de información. Sin embargo, como se verá a continuación, se trata de un sistema tan flexible que es capaz de modelar cualquier tipo de objeto, concreto o abstracto, del mundo real.

RDF es, según la definición del manual oficial publicado por el World Wide Web Consortium (W3C), “un lenguaje para representar información sobre recursos en el World Wide Web” (4). Sin embargo, tal como se apostilla en el mismo documento: “Generalizando el concepto de ‘recurso Web’, RDF también se puede emplear para representar información sobre cosas que puedan *identificarse* en la Web, aun cuando no pueda *accederse* a ellas directamente en la Web”. De esta manera, en la práctica puede aplicarse prácticamente a cualquier objeto, persona o concepto del mundo real, siempre y cuando exista un recurso Web que lo identifique. No se limita pues a páginas web: la frutería de la esquina, las variedades de manzanas que allí se venden o la misma autora de este texto, todo ello y las interacciones entre cada elemento, puede representarse mediante RDF. Con esta modelización de la realidad (o de una parte de ella), las aplicaciones informáticas capaces de leer RDF pueden deducir e intercambiar información sobre aquello que se modela.

La forma en que RDF realiza esta modelización es muy sencilla: la información se descompone en fragmentos, expresados todos ellos siguiendo el mismo esquema básico y extensible, que los dota de semántica interpretable por un ordenador.

Ese esquema básico es una expresión de la forma sujeto-predicado-objeto, donde el predicado (también conocido como “propiedad”) representa la relación que une los otros dos elementos, y el objeto representa el “valor” que adquiere la propiedad simbolizada por el predicado. Cada uno de los tres componentes de la expresión puede, a su vez, formar parte de nuevas expresiones, formando así una red (o grafo) distribuida de elementos interconectados. Lo mejor será ilustrarlo con un ejemplo:

La dueña de la frutería de la esquina mencionada anteriormente, María, es vecina de Juan, el autor de la página Web <http://www.alpanpanyalvinovino.es>. Ambos

viven en la Calle de la Ilusión, número 5, y Juan come muchas manzanas golden, que precisamente se venden en la frutería de María.

Veamos cómo se podría representar esto en expresiones sujeto-predicado-objeto:

SUJETO	PREDICADO	OBJETO
Frutería de la esquina	Pertenece a	María
Juan	Es autor de	http://www.alpanpanyalvinovino.es
María	Vive en	Calle Ilusión, 5
Juan	Vive en	Calle Ilusión, 5
Juan	Come muchas	Manzanas golden
Frutería de la esquina	Vende	Manzanas golden

Tabla 2: Lista de expresiones sujeto-predicado-objeto

Cada línea de la lista anterior es una expresión, afirmación o tripleta, y los sujetos, predicados y objetos son nombres que se utilizan para referirse a objetos, personas o conceptos del mundo real. A esos objetos, personas o conceptos se los llama recursos, entidades o nodos (27). Gráficamente (de forma simplificada) podría dibujarse así:

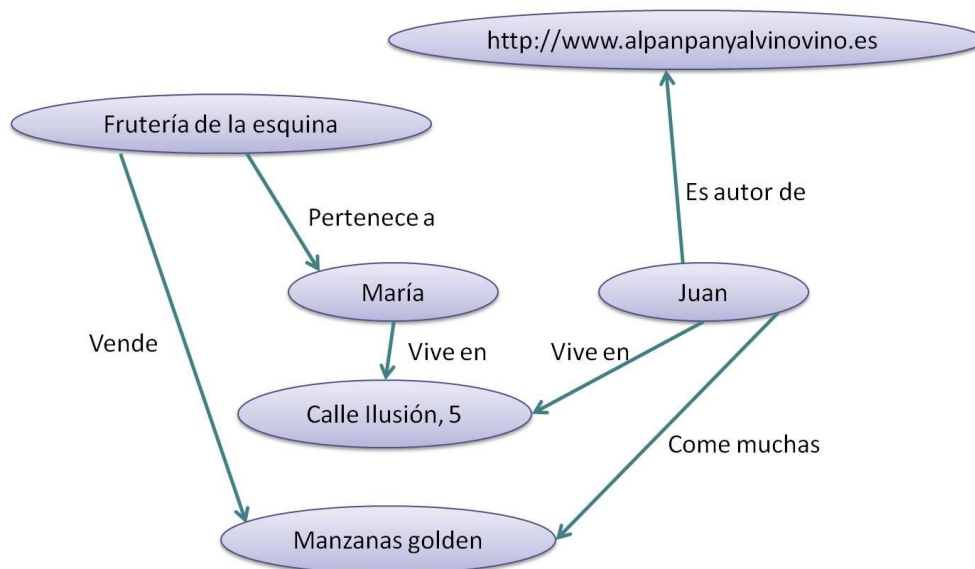


Ilustración 11: Ejemplo de grafo

Como se puede observar en la anterior serie de expresiones, absolutamente todo lo representado tiene un significado, trasladable al mundo real. También se pueden deducir relaciones no explicitadas en las afirmaciones. Por ejemplo, que el autor de la página web <http://www.alpanpanyalvinovino.es> come muchas manzanas golden, o que una vecina de la Calle de la Ilusión nº 5 posee una frutería.

Puede parecer que la información deducida implícitamente de lo anterior no es excesivamente útil, o que en cualquier caso podría haberse hecho algo similar con una

base de datos relacional. Pero, aparte de otras consideraciones que se estudiarán más adelante, RDF lleva esta capacidad de inferencia un paso más allá, gracias al empleo de los llamados Identificadores Universales de Recursos, o URI, por sus siglas en inglés.

Los URI identifican global, única e inequívocamente aquello a lo que representan, de modo que siempre que se emplean en un documento o base de datos RDF, en cualquier lugar del mundo, se refieren al mismo recurso (es decir, el mismo objeto, persona o concepto). Los URI pueden tener el formato de una URL o página web (por ejemplo, <http://purl.org/dc/elements/1.1/title>), pero también puede consistir en un URN (para identificar, por ejemplo, libros por su ISBN, como en <urn:isbn:0860074382>) o una etiqueta o TAG, que puede emplearse para cualquier recurso (<tag:worddb.com,es:palabra>). En realidad, puntualizando un poco más, el identificador de cada recurso individual es la “referencia URI” o URIref. Cada URIref se compone de una URI base, más un identificador opcional al final.

Siguiendo con la analogía de la manzana golden, si María colgara en internet un catálogo de las frutas disponibles en su tienda, y un nutricionista detallara en una página web RDF todas las propiedades nutricionales de cada fruta, incluidas las manzanas golden, y ambos se refirieran a ellas mediante el mismo URIref, Juan podría conocer (empleando un navegador adaptado) la información nutricional de su fruta preferida accediendo desde la misma página de la frutería de María.

De esta forma, se observa que mediante RDF se puede deducir información nueva a partir de información no incluida en ningún documento individual. Esto se debe a que los documentos están enlazados entre sí a través de los vocabularios que comparten, a que además cualquier documento puede hacer uso de cualquier vocabulario existente, y a que los recursos a los que se refieren tienen un identificador universal.

Una última anotación que merece la pena reseñar aquí es qué engloba exactamente el concepto de “recurso”. Tanto el sujeto como el objeto de la tripleta son recursos, pero también el predicado lo es, y por tanto puede aparecer en otra afirmación como sujeto u objeto. En este sentido, la representación de RDF en un grafo es incompleta, ya que pierde la posibilidad de plasmar esas afirmaciones. Empleando tripletas, se podría decir:

SUJETO	PREDICADO	OBJETO
María	Es hermana de	Alejandro
Es hermana de	Es relación de	Parentesco

Tabla 3: Ejemplo de expresiones RDF

En el ejemplo anterior se observa que RDF puede emplearse, además de para describir la relación entre María y Alejandro, para describir el mismo vocabulario empleado para describir dicha relación. Es decir, un vocabulario RDF utiliza precisamente RDF para definirse a sí mismo y sus elementos.

Esta concepción de los predicados como recursos tiene repercusiones más allá de la especificación de un vocabulario. Al requerir RDF que todos los recursos se correspondan con un URI, podemos emplear en nuestro modelo predicados ya existentes y utilizados en otros vocabularios. De hecho, para facilitar la inferencia de las distintas aplicaciones y el intercambio de información, conviene utilizar, siempre que sea posible, predicados que existan ya en lugar de crear uno nuevo, siempre y cuando se desee plasmar el mismo tipo de relación. El buscador Swoogle (28) es una buena herramienta para buscar ontologías y predicados ya existentes.

Además de los las tripletas, los recursos y los URI, existen otros dos elementos de RDF que aún no se han presentado en este documento. Estos elementos son los nodos en blanco y los valores literales. Los nodos en blanco o anónimos son nodos no etiquetados, bien porque no se conoce el URIref del recurso, bien porque no existe ni conviene crearlo. Su uso puede ser útil para representar conceptos que son en realidad una agregación de otros, por ejemplo, una dirección compuesta de calle, número, código postal y municipio.

Otro elemento muy utilizado en RDF es el valor literal. El valor literal consiste en una cadena de caracteres, y puede emplearse exclusivamente como objeto en una tripleta, en lugar de un URI. Los valores literales no se corresponden con ningún recurso, son simplemente texto plano. Se utilizan comúnmente para expresar atributos, tales como el nombre de una persona, el título de un documento, etcétera.

Se resumirá para el lector en la tabla 4 la estructura de una expresión RDF, con una tabla que contiene los distintos elementos que pueden ocupar cada una de las posiciones de la expresión.

SUJETO	PREDICADO (propiedad)	OBJETO (valor)
URI	URI	URI
Nodo en blanco		Nodo en blanco
		Valor literal

Tabla 4: Elementos de una expresión RDF

Para concluir esta sección, se hará un breve resumen de las principales ventajas que aporta RDF frente a otras tecnologías. La más evidente es que RDF está concebido para facilitar la representación distribuida del conocimiento. Toda información publicada en internet en un documento RDF pasa a formar parte del gran grafo global del conocimiento, y abre inmediatamente un abanico de información a aquel que desea consultarla.

Otra ventaja que presenta en relación con las bases de datos relacionales es su flexibilidad. En una base de datos, un cambio en la estructura supone una inversión de tiempo considerable, ya que hay que modificar todo el almacén y reorganizar las claves. Sin embargo, un almacén RDF solo almacena tripletas, que incluso se usan para definir la misma ontología, de modo que basta con borrar, añadir o modificar las tripletas correspondientes. Si en algún momento se desea ampliar el vocabulario e incluir relaciones adicionales, no es necesario hacer ninguna modificación, sin más que introducir las nuevas tripletas en el almacén. No obstante, la rapidez de consulta en una base de datos relacional es superior a la de los almacenes RDF, de modo que habrá que valorar qué factores tienen más peso a la hora de elegir entre uno y otro sistema.

Varios de los inconvenientes del XML que se mencionaron en el apartado anterior no se presentan cuando se emplea RDF. Entre otras diferencias, podríamos destacar:

- En RDF, aunque se desconozca la especificación del vocabulario, se pueden identificar las relaciones y por lo tanto la información semántica de un documento.
- Las consultas se basan siempre en tripletas o combinaciones de tripletas, que solo pueden representarse de una forma, por lo que se simplifica bastante el proceso.
- XML limita las relaciones entre elementos a árboles ordenados. En RDF se elimina esa limitación, pudiendo establecerse cualquier relación entre cualesquiera dos elementos de un grafo.

- Si los archivos XML están definidos de acuerdo a una especificación de estructura y sintaxis, es decir, de acuerdo a una DTD (definición de tipo de documento) o un esquema XML, el documento no será extensible, o requerirá la aceptación de todos aquellos que han participado en su elaboración (27). Con un archivo RDF, sin embargo, cualquiera puede añadir elementos de un nuevo vocabulario sin alterar la información ya existente ni afectar a los demás usuarios del vocabulario anterior.

Evidentemente, no son todas ventajas: la utilización de RDF, sobre todo por lo que supone emplear (y, para ello, buscar e investigar) los URI y ontologías ya existentes, resulta hoy en día aún poco inmediata e intuitiva, lo que sin duda disuade a muchos posibles usuarios e impide que su uso esté aún tan extendido como sería deseable. No obstante, parece que el interés se mantiene, y que poco a poco van surgiendo aplicaciones que hacen uso de esta tecnología.

Conviene matizar que, en cualquier caso, no se ha pretendido realizar aquí una comparativa pormenorizada de los diversos sistemas de representación de la información existentes, puesto que no es el objetivo de este proyecto. Aunque la autora no ha realizado un estudio en profundidad al respecto, un análisis superficial de las diversas tecnologías le ha permitido juzgar como adecuado el empleo de RDF para modelar la información que ocupa a este trabajo.

3.2.2 Serialización: Notation 3

Actualmente existen varios formatos para representar formalmente un modelo RDF. Los más conocidos son RDF/XML, Notation 3, Turtle y N-Triples (29). Los dos últimos se derivan del anterior, y son versiones más simplificadas y reducidas de este. Dado que todos se emplean para representar un mismo modelo abstracto, la única diferencia real entre ellos es el formato que utilizan para plasmarlo.

Pese a que probablemente la notación más extendida para serializar un modelo RDF sea RDF/XML, en este documento solo se comentará que este formato hace uso de la sintaxis XML para representar el modelo.

La notación que se describirá en este apartado es Notation 3 (o N3) (30). El motivo de obviar RDF/XML, un formato empleado por buena parte de los desarrolladores, es doble: por un lado, N3 resulta más escueto y, sobre todo, más legible para una persona. Por otro, el lenguaje de consultas que se empleará en la aplicación desarrollada, Protocol and RDF Query Language (SPARQL), utiliza una

sintaxis muy similar a la de este formato, de modo que al familiarizar al lector con él, se le facilitará también la asimilación posterior de SPARQL.

Recordando que la estructura básica de RDF es la tripleta sujeto-predicado-objeto, y que los tres elementos deben ser URI, una tripleta sencilla en N3 se representaría de la siguiente forma:

```
<#Laura> <#conoce_a> <#Juan> .
```

Se observa que cada recurso se escribe entre los símbolos “<” y “>”, y que la afirmación va seguida de un punto. El URI puede escribirse completo (por ejemplo, <http://xmlns.com/foaf/spec/#term_knows>), o acortarlo para facilitar la legibilidad. En el ejemplo anterior, si simplemente se precede el nombre de #, significa que se identifica el recurso únicamente en el documento actual.

Para hacer referencia a un URI externo al documento, se puede escribir entero, o abreviar usando los espacios de nombres, concepto prestado de XML. Así, se declara el espacio de nombres al comienzo del documento, y su abreviatura se puede emplear a lo largo de éste en su lugar, seguida de dos puntos. Para declarar el espacio de nombres en N3, se utiliza la directiva @prefix, como en:

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .  
@prefix alum: <http://www.ejemplo.org/alumnos#> .  
  
alum:Laura_Sanchez foaf:knows alum:Juan_Garcia .
```

Para obtener el URIref completo en el ejemplo anterior, basta con sustituir la abreviatura por su URI. De este modo, el URIref de Laura sería: <http://www.ejemplo.org/alumnos#Laura>. En este ejemplo se ha hecho uso de un vocabulario ya existente, creado para describir relaciones entre personas: Friend Of A Friend (FOAF).

Falta aún mostrar cómo se escribirían los nodos en blanco y los valores literales. Los identificadores para escribir los nodos en blanco tienen el mismo formato que los empleados con los espacios de nombres, sin más que usar “_” como espacio de nombres. Se puede emplear cualquier nombre después de los dos puntos, pero hay que recordar que debe usarse el mismo a lo largo de todo el documento siempre que se haga referencia al mismo recurso. Por su parte, los valores literales se escriben simplemente entre comillas.

Para añadir a lo anterior la afirmación de que Laura Sánchez, con URI `<http://www.ejemplo.org/uc3m/alumnos#Laura_Sanchez>`, conoce a una tal Sandra Fernández, de la que se desconoce si tiene URI, se podría escribir:

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix alum: <http://www.ejemplo.org/alumnos#> .

alum:Laura_Sanchez    foaf:knows    alum:Juan_Garcia .
alum:Laura_Sanchez    foaf:knows    _:amigal .
_:amigal              foaf:name      "Sandra Fernández" .
```

Existen más herramientas en N3 que se pueden utilizar para abreviar la representación de un modelo RDF, siempre en aras de la simplicidad. Dos de estas herramientas son la coma y el punto y coma. Cuando se repiten en varias expresiones el mismo sujeto y predicado, separar los objetos con comas permite la omisión de estos dos elementos. Así, el hecho de que Laura Sánchez conoce a Juan García y a Sandra Fernández se podría escribir de la siguiente forma:

```
alum:Laura_Sanchez    foaf:knows    alum:Juan_Garcia ,
_:amigal              foaf:knows    _:amigal .
_:amigal              foaf:name      "Sandra Fernández" .
```

Igualmente, si lo que se repite es el sujeto, utilizando el punto y coma se evitará la necesidad de duplicarlo, como en el siguiente ejemplo:

```
_:amigal              foaf:name      "Sandra Fernández" ;
                      foaf:knows    alum:Laura_Sanchez .
```

Con los conceptos presentados en el presente apartado se ha pretendido dar al lector una visión global de la notación N3 y dotarle de los conceptos necesarios para seguir los capítulos siguientes, más que ofrecerle una especificación detallada y pormenorizada de esta notación.

3.2.3 Ontologías

En este documento se han venido haciendo varias menciones a los *vocabularios* RDF, en alusión a la especificación de los términos y relaciones que se emplean para modelar determinado dominio. Aunque “vocabulario” es un término que se utiliza habitualmente en este contexto, un concepto de uso tanto o más extendido que el anterior, para referirse esencialmente a lo mismo (si bien con distintos matices) es el de *ontología*.

DEFINICIÓN

Aunque existen múltiples acepciones para la palabra “ontología”, aquí se expondrá una de las definiciones formales más aceptadas, dada por Gruber en (31). La definición propuesta en dicho artículo fue: “Una ontología es una especificación explícita de una conceptualización”, que posteriormente fue matizada por Borst (32), quedando así: “Una ontología se define como una especificación formal de una conceptualización compartida”.

Estas definiciones se pueden explicar atendiendo a los distintos vocablos que las integran. Así, por “especificación explícita” se entiende que el conjunto de conceptos, términos y propiedades que lo componen se definen explícitamente, así como las restricciones aplicables a estos. “Formal” se refiere al hecho de que la ontología debe poder ser leída por una máquina. “Conceptualización” significa un modelo abstracto de algún fenómeno del mundo, habiendo identificado los conceptos relevantes de dicho fenómeno, y “compartida” indica que el conocimiento debe ser consensuado por un grupo, y no provenir de un único individuo.

MOTIVOS PARA USAR ONTOLOGÍAS

Hay muchas razones que pueden motivar el desarrollo de una ontología, tal como se expone en (33). Entre otras, destacan:

- Compartir una interpretación común de la estructura de la información entre varias personas o agentes. Este aspecto se ha mencionado ya en los apartados anteriores, ya que con una ontología común (o con partes de esta), las aplicaciones pueden extraer o combinar información desde diferentes fuentes.
- Permitir la reutilización del conocimiento de los dominios. Creada una ontología para determinado dominio, otras personas podrán utilizarla para expresar nueva información relativa al mismo dominio, o a uno que lo integre.
- Hacer explícitas las suposiciones sobre el dominio. Así, si cambia el conocimiento del dominio, se podrán cambiar fácilmente las suposiciones. Si, por el contrario, se integran en la programación de la aplicación, será menos evidente y resultará difícil buscar, comprender y cambiar las suposiciones.

- Separar el conocimiento del dominio del conocimiento operativo. Si la ontología se mantiene independiente de los componentes específicos del dominio, se favorece su reutilización en distintos dominios.

PRINCIPIOS DE DISEÑO

Conviene tener en cuenta una serie de principios que proporcionarán estabilidad y calidad de la ontología desarrollada, sintetizados en (34):

- Claridad y objetividad: mediante definiciones objetivas de los términos.
- Integridad: mediante definiciones completas.
- Coherencia: para permitir deducciones coherentes con las definiciones.
- Maximización de la extensibilidad monotónica: que asegure que los nuevos términos no modifiquen la definición de los ya existentes.
- Minimización de los compromisos ontológicos: reduciendo al máximo las aseveraciones sobre el mundo que se modela, para permitir que los usuarios empleen la ontología haciendo las especializaciones necesarias.
- Principio de Distinción Ontológica: no debe haber intersección entre las clases que componen la ontología.
- Diversificación de jerarquías para favorecer los mecanismos de herencia múltiple.
- Modularidad.
- Minimización de la distancia semántica entre conceptos próximos.
- Estandarización de los nombres cuando sea posible.

COMPONENTES DE UNA ONTOLOGÍA

Los distintos elementos que componen una ontología son:

- **Clases** o conceptos: representan una categoría o conjunto de individuos. Pueden ser conceptos concretos o abstractos, reales o ficticios. Ejemplo: “persona”, “félido”, “automóvil”.
- **Relaciones** o propiedades: representan un tipo de relación entre clases del dominio. Ejemplo: “parentesco”.
- **Instancias**: son los elementos individuales que componen los conjuntos definidos por las clases. Ejemplo: “María”, miembro de la clase “persona”.
- **Restricciones** en las relaciones. Ejemplo: Si “María” participa en una relación de “parentesco”, esta solo puede producirse con miembros de la clase “persona”.

Las ontologías más avanzadas pueden contener también otros elementos, como funciones y axiomas, que no se detallarán aquí, al quedar fuera de la esfera de aplicación de este documento.

METODOLOGÍA DE DESARROLLO

Partiendo de la consideración de que hay muchas formas correctas de modelar un dominio determinado, se especifican aquí brevemente los pasos propuestos en (33) para crear una ontología. En este texto se adaptan ligeramente para las particularidades de RDF:

1. **Determinar el dominio y el ámbito de la ontología.** A través, por ejemplo, de la definición del dominio, las partes que lo componen y el propósito de la aplicación.
2. **Estudiar el uso de ontologías ya existentes.** Ello permitirá el intercambio de información y la consulta entre distintas fuentes.
3. **Enumerar términos importantes.** Suponiendo que no se reutiliza una ontología existente, debe elaborarse una lista de términos importantes en el dominio, sin categorizarlos ni ubicarlos aún en una jerarquía.
4. **Definir las clases y la jerarquía.** Se pueden definir primero las clases más genéricas e ir especificando las distintas subclases, o a la inversa, creando superclases a partir de las clases de nivel más bajo, o bien adoptar una estrategia mixta, combinando las dos anteriores.
5. **Definir las propiedades y relaciones.** Las propiedades de las clases, que definen su estructura interna, pueden considerarse relaciones con otros elementos, tal como sucede en RDF.
6. **Especificar las restricciones de las relaciones.** Se especifican restricciones como cardinalidad o tipo del dominio y del ámbito, entendiéndose por dominio el conjunto de clases que pueden ser sujeto de la relación, y ámbito aquellas que pueden ser el objeto de esta.
7. **Crear instancias.** Se crean instancias individuales de las clases desarrolladas.

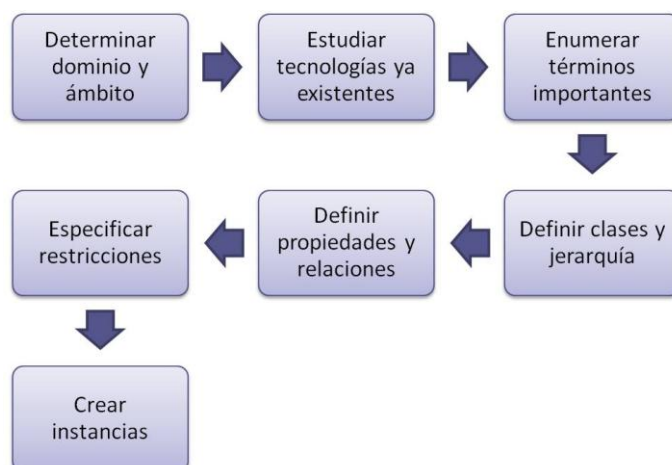


Ilustración 12: Metodología de desarrollo de ontologías

Tras esta introducción teórica a las ontologías y su desarrollo, se explicará el lenguaje que se va a emplear para la definición de la ontología que se construirá en el presente proyecto: el RDF Schema.

3.2.4 RDF Schema

RDF Schema (4) es un lenguaje normalizado por el W3C para la definición de ontologías RDF. Mediante su empleo, se obtiene precisamente un grafo RDF que representa la ontología descrita.

Se utilizarán aquí indistintamente “propiedad” y “predicado” para referirse a las relaciones que unen las distintas clases. Asimismo, se hará uso de los habituales espacios de nombres `rdf` y `rdfs`, para referirse respectivamente a la URI del vocabulario RDF y a la del RDF Schema:

```
rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
rdfs="http://www.w3.org/2000/01/rdf-schema#"
```

DESCRIPCIÓN DE CLASES

Para describir clases se utilizan los recursos de RDF Schema `rdfs:Class` y `rdfs:Resource`, y las propiedades `rdf:type` y `rdfs:subClassOf`.

Una clase será, así, todo recurso que tenga una propiedad `rdf:type` con valor `rdfs:Class`.

Para establecer una jerarquía que exprese la especialización entre clases, se utiliza la propiedad `rdfs:subClassOf`. Esta propiedad es transitiva, por lo que si `c` es subclase de `b`, y `b` es subclase de `a`, `c` también es subclase de `a`.

Conviene señalar que, aunque un recurso sea subclase de otro, y por tanto pueda inferirse que se trate de una clase, el código de buenas prácticas recomienda proporcionar esta información explícita en el documento.

Un ejemplo sencillo de lo explicado hasta el momento podría contener las siguientes expresiones:

```
ex:Persona      rdf:type      rdfs:Class .
ex:Mujer        rdf:type      rdfs:Class .
ex:Mujer        rdfs:subClassOf ex:Persona .
```

DESCRIPCIÓN DE PROPIEDADES

Para describir las propiedades que caracterizan a las clases, se utilizan la clase de RDF `rdf:Property` y las propiedades de RDF Schema `rdfs:domain`, `rdfs:range` y `rdfs:subPropertyOf`.

Todas las propiedades se describen como instancias de la clase `rdf:Property`, y son, por defecto, de aplicación global. Para describir cómo deben combinarse propiedades y clases en un vocabulario RDF determinado, RDF Schema proporciona dos elementos restrictivos: `rdfs:range` y `rdfs:domain`.

La propiedad de *ámbito*, `rdfs:range` indica que los valores que toma el objeto de determinado predicado son instancias de la clase señalada. Si un predicado (o propiedad) tiene más de un ámbito definido, los objetos de ese predicado deben ser instancias de todas las clases especificadas como ámbitos.

El ámbito de una propiedad puede ser un valor literal, que pertenecería a la clase `rdfs:Literal`.

La propiedad de *dominio*, `rdfs:domain`, indica que un determinado predicado solo puede aplicarse a las clases por él especificadas. Tal como ocurría con el ámbito, si para un predicado se especifican varios dominios, el sujeto al que se aplique ese predicado debe ser una instancia de todas las clases especificadas en los dominios.

Finalmente, las propiedades, al igual que las clases, pueden especializarse mediante la propiedad `rdfs:subPropertyOf`. Todas las propiedades `rdfs:range`

y `rdfs:domain` que se aplican a una propiedad, se aplican también a sus subpropiedades herederas.

A modo de resumen, se aglutinan en el siguiente ejemplo todos los conceptos explicados en este subapartado:

```
ex:pariente      rdf:type      rdf:Property .
ex:madreDe       rdf:type      rdf:Property .
ex:madreDe       rdfs:subPropertyOf ex:pariente .
ex:madreDe       rdfs:range     ex:Persona .
ex:madreDe       rdfs:domain    ex:Mujer .
```

INFORMACIÓN ADICIONAL

Para finalizar, se presentan aquí otras propiedades incluidas en RDF Schema que pueden proporcionar información complementaria sobre un vocabulario o sobre las instancias individuales:

- Propiedad `rdfs:comment`. Permite describir un recurso de forma que sea legible para una persona.
- Propiedad `rdfs:label`. Permite proporcionar una versión legible del nombre de un recurso.
- Propiedad `rdfs:seeAlso`. Se utiliza para indicar un recurso que contenga información adicional sobre el recurso representado por el sujeto.

3.2.5 SPARQL

SPARQL (35) es un lenguaje de consulta para grafos RDF, recomendado oficialmente por el W3C.

Puesto que el lector ya está familiarizado con la notación N3, aquí se realizará una presentación más somera de la sintaxis de este lenguaje (similar a N3, aunque con variaciones), ilustrándola simplemente con un ejemplo que contiene todos los elementos clave.

Los componentes de una consulta SPARQL son los siguientes:

- Declaraciones de prefijos, para abreviar los URI.
- Una expresión de resultados, que identifica la información (mediante variables) que se solicita en la consulta.
- Definiciones del grafo o grafos en los que se realiza la consulta.
- Un patrón de consulta, qué especifica qué buscar en el grafo RDF.
- Modificadores de consulta, que reorganizan o limitan los resultados obtenidos.

A continuación, y para finalizar esta breve exposición, se muestra la sintaxis por medio de un ejemplo de consulta más o menos completa:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX ex: <http://example.org/properties/1.0#>

SELECT DISTINCT ?person ?name ?age

FROM <http://rdf.example.org/personA.rdf>
FROM <http://rdf.example.org/personB.rdf>

WHERE { ?person      rdf:type    foaf:Person ;
        {            foaf:name  ?name.      }
        UNION
        {            rdfs:label ?name.      }
OPTIONAL { ?person ex:age ?age } .
FILTER (!REGEX(?name, "Bob"))
}
ORDER BY ASC(?name) LIMIT 10
```

Que, comentada (anteponiendo almohadillas a los comentarios para que el formato de la consulta siga siendo correcto) y resaltando las observaciones en color azul para mejorar la legibilidad, queda como sigue:

```
# Declaraciones de prefijos:
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX ex: <http://example.org/properties/1.0#>

# La expresión de resultados.
# Las variables se especifican mediante ?
SELECT DISTINCT ?person ?name ?age

# Se definen los grafos que se consultarán
FROM <http://rdf.example.org/personA.rdf>
FROM <http://rdf.example.org/personB.rdf>

# Se especifica el patrón de consulta, entre corchetes.
# Incluye la unión de dos posibles variaciones en el grafo
# que representen lo mismo:
WHERE { ?person      rdf:type    foaf:Person ;
        { foaf:name  ?name.      }
        UNION
        { rdfs:label ?name.      }

# También incluye una parte para datos que no tienen
# por qué estar:
OPTIONAL { ?person ex:age ?age } .

# Se filtra la variable ?name mediante una expresión regular
# para que solo se obtengan los resultados que coincidan
FILTER (!REGEX(?name, "Bob"))

# Se cierran los corchetes con el patrón de consulta:
}

# Se ordenan los resultados en orden ascendente,
# con un límite de 10 coincidencias:
ORDER BY ASC(?name) LIMIT 10
```

4. Planificación del proyecto

En este apartado se desglosarán las distintas fases del proyecto y se estimará el esfuerzo necesario para llevarlas a cabo. Mediante este análisis se podrá evaluar la viabilidad económica y realizar un presupuesto adecuado para el proyecto.

Se comenzará identificando las tareas en que se divide el proyecto, para posteriormente realizar un plan de trabajo, trazar un diagrama de Gantt y finalizar con un presupuesto ajustado a las estimaciones realizadas.

4.1 Tareas a realizar

- **Estudio preliminar:**
 - Estado de la cuestión: se evalúan las herramientas actuales para la traducción de subtítulos.
 - Aprendizaje de tecnologías: la autora ha de dedicar el tiempo necesario a familiarizarse con el entorno y el lenguaje de programación elegido, así como con la modelización del conocimiento y las ontologías.
 - Planteamiento del problema: se especifica cuál es el problema y qué objetivos debe cumplir la solución propuesta.
- **Análisis y diseño:**
 - Especificación de requisitos: se definen los requisitos funcionales y no funcionales que debe satisfacer la aplicación.
 - Diseño: se realizan los diagramas UML (Unified Modeling Language) (36) correspondientes y se determinan la solución y la arquitectura que se adoptará.
- **Desarrollo:**
 - Instalación del software: se instalan y se configuran la base de datos y el SDK Visual C# Express.
 - Desarrollo de la interfaz gráfica de usuario:
 - Pantalla contenedora Multiple Document Interface (MDI).
 - Pantalla de edición de diccionario.
 - Pantalla de consulta de diccionario.
 - Control de usuario: subtítulo
 - Pantalla de edición y traducción de subtítulos.
 - Ventanas y clases secundarias.
 - Localización de la aplicación (inglés y castellano)

- Persistencia de configuración de usuario
- Desarrollo de la capa de presentación:
 - Presentador base y clases auxiliares.
 - Clase de presentación de editor de diccionario.
 - Clase de presentación de diccionario.
 - Clase de presentación de subtítulos.
- Desarrollo de la capa de negocio:
 - Clase de negocios Base y de la clase de validación.
 - Clase de entrada básica.
 - Clase de equivalencia semántica.
 - Clase de entrada extendida.
 - Clase de subtítulo y colección.
- Desarrollo de la capa de servicio.
- Desarrollo de la capa de acceso a datos:
 - Instalación de la librería SemWeb.NET (38) y creación del sumidero de resultados.
 - Conversión de los objetos de negocios a tabla de datos.
 - Métodos de consulta y modificación de la base de datos.
- Pruebas de la aplicación: se comprueba que se cumplen todos los requisitos fijados con anterioridad.
- **Documentación:**
 - Memoria del proyecto: incluye todos los apartados pertinentes: introducción, estado de la cuestión, representación del conocimiento, etcétera.
 - Presentación del proyecto.

4.2 Diagrama de Gantt

Una vez definidas las tareas que deben llevarse a cabo, se procede a planificar su realización. La dedicación media diaria será de 3 horas al día, de lunes a sábado.

	Task Name	Duration	Start	Finish
1	- Estudio preliminar	22 days	Wed 23/09/09	Sat 17/10/09
2	Estado de la cuestión	5 days	Wed 23/09/09	Mon 28/09/09
3	Aprendizaje de tecnologías	14 days	Tue 29/09/09	Wed 14/10/09
4	Planteamiento del problema	3 days	Thu 15/10/09	Sat 17/10/09
5	- Análisis y diseño	15 days	Mon 19/10/09	Wed 04/11/09
6	Análisis y especificación de requisitos	5 days	Mon 19/10/09	Fri 23/10/09
7	Diseño	10 days	Sat 24/10/09	Wed 04/11/09
8	- Desarrollo	161 days	Thu 05/11/09	Tue 11/05/10
9	Instalación del software	3 days	Thu 05/11/09	Sat 07/11/09
10	- Desarrollo de la interfaz gráfica	57 days	Mon 09/11/09	Wed 13/01/10
11	Pantalla contenedora MDI	12 days	Mon 09/11/09	Sat 21/11/09
12	Pantalla de edición de diccionario	11 days	Mon 23/11/09	Fri 04/12/09
13	Pantalla de consulta de diccionario	6 days	Sat 05/12/09	Fri 11/12/09
14	Control de usuario: subtítulo	6 days	Sat 12/12/09	Fri 18/12/09
15	Pantalla de traducción de subtítulos	12 days	Sat 19/12/09	Fri 01/01/10
16	Ventanas y clases secundarias	8 days	Sat 02/01/10	Mon 11/01/10
17	Localización de la aplicación	2 days	Tue 12/01/10	Wed 13/01/10
18	- Desarrollo de la capa de presentación	37 days	Thu 14/01/10	Thu 25/02/10
19	Presentador base y clases auxiliares	7 days	Thu 14/01/10	Thu 21/01/10
20	Presentación del editor del diccionario	9 days	Fri 22/01/10	Mon 01/02/10
21	Presentación del diccionario	6 days	Tue 02/02/10	Mon 08/02/10
22	Presentación de subtítulos	10 days	Tue 09/02/10	Fri 19/02/10
23	Pruebas	5 days	Sat 20/02/10	Thu 25/02/10
24	- Desarrollo de la capa de negocio	18 days	Fri 26/02/10	Thu 18/03/10
25	Clase de negocios base y clase de validación	4 days	Fri 26/02/10	Tue 02/03/10
26	Entrada básica	3 days	Wed 03/03/10	Fri 05/03/10
27	Equivalencia semántica	3 days	Sat 06/03/10	Tue 09/03/10
28	Entrada extendida	2 days	Wed 10/03/10	Thu 11/03/10
29	Subtítulo y colección	3 days	Fri 12/03/10	Mon 15/03/10
30	Pruebas	3 days	Tue 16/03/10	Thu 18/03/10
31	Desarrollo de la capa de servicio	2 days	Fri 19/03/10	Sat 20/03/10
32	- Desarrollo de la capa de acceso a datos	24 days	Mon 22/03/10	Sat 17/04/10
33	Librería SemWeb y sumidero de resultados	3 days	Mon 22/03/10	Wed 24/03/10
34	Conversión de objetos a tabla de datos	4 days	Thu 25/03/10	Mon 29/03/10
35	Métodos de consulta y modificación de la BD	12 days	Tue 30/03/10	Mon 12/04/10
36	Pruebas	5 days	Tue 13/04/10	Sat 17/04/10
37	Pruebas de la aplicación	20 days	Mon 19/04/10	Tue 11/05/10
38	- Documentación	43 days	Wed 12/05/10	Wed 30/06/10
39	Memoria del proyecto	35 days	Wed 12/05/10	Mon 21/06/10
40	Presentación	8 days	Tue 22/06/10	Wed 30/06/10

Ilustración 13: Cuadro de tareas del diagrama de Gantt

HERRAMIENTA PARA LA TRADUCCIÓN ASISTIDA EN LA INDUSTRIA AUDIOVISUAL

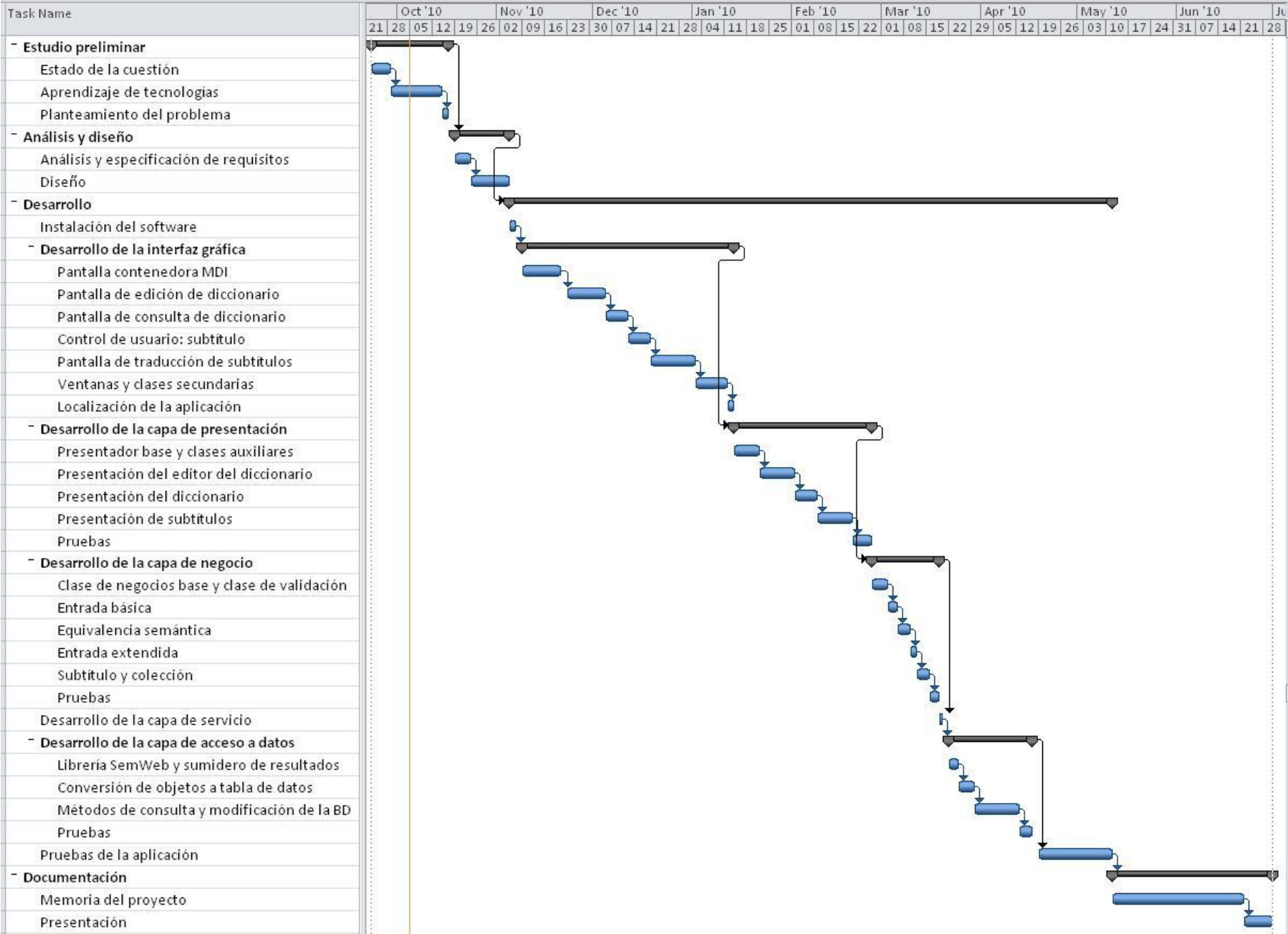


Ilustración 14: Diagrama de Gantt

4.3 Presupuesto

Para realizar un presupuesto lo más preciso posible, se procederá a dividir el proyecto en cada fase que lo compone, como análisis, diseño o implementación.

Para cada fase se evaluará la dedicación de personal asociada, lo que permitirá calcular los gastos directos de personal. Posteriormente se estimarán el resto de gastos directos, como infraestructuras o material fungible.

4.3.1 Desglose del proyecto en fases

En primer lugar, se delimitan las distintas fases del proyecto y las horas requeridas para completar cada una. Para ello, la tabla de tareas del diagrama de Gantt proporcionará toda la información necesaria, que se resume en la siguiente tabla.

DURACIÓN DEL PROYECTO		
Fase del proyecto	Duración (días)	Duración (horas)
Estudio preliminar	8	24
Análisis	5	15
Diseño	10	30
Implementación	141	423
Pruebas	20	60
Documentación	43	129
Total		660

Tabla 5: Duración del proyecto por fases

Como se puede apreciar en el anterior desglose, se ha omitido el período destinado a formación de la autora, por no ser un gasto imputable al desarrollo del proyecto.

Una vez estimada la dedicación total necesaria, se calcularán los recursos humanos destinados a la elaboración del proyecto.

4.3.2 Gastos de personal

Para poder evaluar los gastos de recursos humanos a repercutir en la realización del proyecto, habrán de tenerse en cuenta las categorías profesionales de cada miembro del personal que ha participado en su elaboración.

Al tratarse de un presupuesto ficticio, se recurre a la herramienta en línea Infojobs Trends Salarios (39) para obtener unos sueldos de referencia, que serán los que se apliquen en esta estimación de gastos.

SALARIOS		
	Sueldo bruto anual	Retribución horaria
Jefe de proyecto	38.000 €	21,59 €
Analista	29.000 €	16,48 €
Programador	20.000 €	11,36 €

Tabla 6: Salarios

Para calcular la retribución horaria se ha dividido el sueldo bruto anual entre 1.760 horas de trabajo efectivo al año, tal como contemplan los convenios de trabajadores. Ello supondría 11 meses de trabajo real, a una media de 160 horas de trabajo al mes.

A continuación se estimará la dedicación de cada categoría profesional a cada actividad que compone el proyecto.

DEDICACIÓN DE PERSONAL POR FASE DE PROYECTO				
Fase del proyecto \ Cargo	Jefe de proyecto	Analista	Programador	Total fase
Estudio preliminar	24			24
Análisis	7,5	7,5		15
Diseño	15	15		30
Implementación		141	282	423
Pruebas			60	60
Documentación		129		129
Total	46,5	292,5	342	660

Tabla 7: Dedicación de personal por fase de proyecto (en horas)

Y por último, se calculará el coste destinado a recursos humanos en la totalidad del proyecto, en función de las distintas categorías profesionales.

COSTES DE PERSONAL			
	Horas trabajadas	Retribución horaria	Coste total
Jefe de proyecto	46,5	21,59 €	1.003,94 €
Analista	292,5	16,48 €	4.820,40 €
Programador	342	11,36 €	3.885,12 €
Total			9.709,46 €

Tabla 8: Costes de personal

4.3.3 Gastos directos materiales

Para elaborar un presupuesto preciso, se incluirán los gastos de bienes materiales en los que se ha incurrido. El gasto del ordenador portátil se incluirá en la proporción en que se va a destinar su uso a la elaboración de este proyecto, en cuanto a la vida útil estimada del aparato.

Asumiendo una vida útil de 5 años, a una media de 1.760 horas trabajadas al año, la vida útil estimada del portátil asciende a 8.800 horas. Puesto que el tiempo dedicado a la elaboración del proyecto se ha calculado en 660 horas, y el precio del ordenador fue de 1.099 euros, la cantidad imputable al proyecto será de:

GASTOS MATERIALES			
	Cantidad	Precio por unidad	Coste total
Ordenador portátil	0,075	1.099 €	82,43 €
Memoria en CD	2	2 €	4 €
Material de oficina			30 €
Total			116,43 €

Tabla 9: Gastos materiales

4.3.4 Resumen del presupuesto

Al desarrollarse en un entorno académico y ser realizado por una sola persona, el presupuesto es únicamente una aproximación al que se presentaría de elaborarse en el ámbito empresarial. No obstante, dado que se desea que sea lo más realista posible, se incluirá en el documento una partida de gastos indirectos, que recogería todos aquellos no asociados directamente con el proyecto en cuestión, tales como alquiler de oficina o gastos de agua, luz, teléfono o ADSL.

En condiciones normales, para determinar los gastos indirectos se estudiaría el total de gastos indirectos de la empresa en relación con el total de gastos directos en un período determinado, obteniéndose la tasa a aplicar. Puesto que no es posible realizar dicho cálculo, se supondrá un porcentaje del 10% sobre los gastos directos calculados.

El resumen del presupuesto realizado quedará pues como sigue:

PRESUPUESTO DEL PROYECTO	
GASTOS	
<i>Gastos directos</i>	
Personal	9.709,46 €
Materiales	116,43 €
Total de gastos directos	9.825,89 €
<i>Gastos indirectos (10 % del subtotal anterior)</i>	
	982,59 €
Total de gastos	10.808,48 €
<i>Beneficios (30% del subtotal anterior)</i>	
	3.242,54 €
Coste total del proyecto antes de impuestos	14.050,98 €
<i>IVA (16%)</i>	
	2.248,16 €
Total	16.299,14 €

Tabla 10: Presupuesto

5. Análisis del problema

5.1 Planteamiento del problema

Tal como se ha expuesto en los primeros apartados, la traducción de subtítulos está en auge y, aunque las nuevas tecnologías han entrado en el sector con fuerza, no se ha prestado la atención debida al elemento de traducción dentro del proceso de creación de los subtítulos.

Existe una variedad de soluciones profesionales para la creación, pautado, edición, conversión y traducción de subtítulos, pero todas hacen énfasis en los primeros aspectos, quedando la faceta de traducción relegada a un segundo plano. La clienta y autora, traductora audiovisual de subtítulos, desea una solución destinada exclusivamente al traductor, que le proporcione utilidades similares a las que ofrecen otras herramientas CAT, pero adaptada a las particularidades de la traducción de subtítulos (carga de ficheros de subtítulos, integración de vídeo, regla de los seis segundos...).

Asimismo, deben poder mostrarse a la vez dos archivos de subtítulos, el archivo matriz en la lengua original y el archivo en la lengua meta, para poder cotejar el texto en todo momento.

La utilidad de gestión del repositorio terminológico debe facilitar en la medida de lo posible el trabajo del traductor, teniendo en cuenta la heterogeneidad de la terminología empleada. Además de ofrecer las sugerencias automáticamente, debe permitir muy especialmente la inclusión de frases hechas, expresiones coloquiales, modismos, etcétera, que contribuyan a aumentar la riqueza lingüística de la traducción.

Para posibilitar el uso del repositorio almacenado en cualquier momento, se facilitará un sistema independiente de consulta y edición del repositorio, de modo que no será necesario que el traductor esté trabajando en una traducción de subtítulos para poder acceder cómodamente a la terminología almacenada.

5.2 Requisitos fijados

Tras el planteamiento inicial, se realiza una recogida formal de requisitos, que se dividirán en funcionales y no funcionales. El usuario único de la aplicación será el traductor.

Para estructurar la recogida de requisitos, se empleará la siguiente plantilla:

Identificador: <i>Nombre</i>	
Descripción	
Prerrequisitos	
Necesidad	<input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Fuente	

Donde:

- **Identificador:** Identifica unívocamente el requisito. El formato que tendrá es:

RX-YYY

Donde:

- *X*: especifica el tipo de requisito. *F* si el requisito es funcional, *FR* si es funcional de restricción, y *NF* si es no funcional.
- *YYY*: número que lo identifica dentro de una secuencia.
- **Nombre:** texto corto y descriptivo del requisito.
- **Descripción:** explica en pocas líneas en qué consiste el requisito.
- **Prerrequisitos:** cualquier requisito que éste tenga como condición para poder cumplirse.
- **Necesidad:** indica el nivel de obligatoriedad del requisito dentro de la aplicación final. Los valores posibles son: Esencial, Deseable u Opcional. Un requisito será Esencial si el cliente exige su implementación; Deseable si es conveniente, pero no obligatorio, incluirlo y Opcional si su inclusión no es prioritaria.
- **Prioridad:** indica la prioridad del requisito durante el desarrollo. Los valores posibles son: Alta, Media o Baja.
- **Fuente:** indica el origen a partir del cual se ha obtenido el requisito. Las posibles fuentes en este proyecto son el cliente y el analista.

5.2.1 Requisitos funcionales

Funciones que debe ser capaz de realizar el sistema, y el comportamiento que ha de tener este en circunstancias particulares. Esto incluye los requisitos de restricción, que especifican los comportamientos no permitidos por el sistema.

RF-001: Consulta en diccionario

Descripción	El usuario puede realizar consultas en el diccionario-repositorio desde una ventana destinada a tal efecto, y también desde la misma ventana de subtitulado. En la consulta se puede especificar la categoría gramatical o el ámbito en que se aplica la relación semántica.		
Prerrequisitos			
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Fuente	Cliente		

RF-002: Navegación por últimas consultas realizadas

Descripción	El usuario puede navegar por las últimas consultas realizadas, tanto en la ventana de consulta como en la de edición del diccionario.		
Prerrequisitos	RF-001: <i>Consulta en diccionario</i> : Deben haberse realizado consultas con anterioridad.		
Necesidad	<input type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input checked="" type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Fuente	Cliente, analista		

RF-003: Consulta no segura en diccionario

Descripción	Desde las ventanas de diccionario y subtitulación, el usuario puede realizar consultas no seguras en el repositorio. Estas consistirán en la búsqueda a lo largo de varios nodos del grafo, en lugar de solo los inmediatamente próximos, para obtener más posibles equivalencias semánticas del término original, y validarlas si las considera adecuadas.		
Prerrequisitos			
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Fuente	Cliente, analista		

RF-004: Agregación de término o expresión al diccionario

Descripción	<p>El usuario puede añadir nuevas entradas al diccionario-repositorio. Los campos que puede contener son:</p> <ul style="list-style-type: none"> • <u>Entrada</u>: texto con el término o expresión. • <u>Categoría gramatical</u>: categoría gramatical de la entrada (sustantivo, verbo, adjetivo, adverbio u otra). • <u>Tipo de expresión</u>: término, locución, modismo o forma compuesta. • <u>Grafía alternativa</u>: otra posible representación gráfica del mismo término (campo optativo). • <u>Idioma</u>: lengua de la entrada. • <u>Comentarios</u>: anotaciones sobre la entrada (campo optativo). • <u>Traducciones</u>: lista de traducciones al segundo idioma de trabajo, con sus correspondientes ámbitos, contextos y ejemplos. • <u>Sinónimos</u>: lista de sinónimos en su idioma, con sus correspondientes ámbitos, contextos y ejemplos.
Prerrequisitos	
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Fuente	Cliente

RF-005: Modificación de entrada en el diccionario

Descripción	<p>El usuario puede editar entradas ya existentes en el diccionario-repositorio, añadiendo relaciones a estas, modificándolas o eliminándolas.</p>
Prerrequisitos	<p>RF-001: <i>Consulta en diccionario</i> RF-004: <i>Agregación de término o expresión al diccionario</i></p>
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja
Fuente	Cliente

RF-006: Carga de archivo de subtítulos

Descripción	El usuario puede abrir archivos de subtítulos para su edición o traducción en la ventana correspondiente. Se pueden abrir en la columna de archivo fuente o en la columna de archivo destino.
Prerrequisitos	
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Fuente	Cliente

RF-007: Creación de archivo de subtítulos

Descripción	El usuario puede crear un nuevo archivo de subtítulos destino a partir de un archivo fuente. El archivo será idéntico en cuanto a número de subtítulos y códigos de tiempo, pero los campos de subtítulos y comentarios estarán vacíos.
Prerrequisitos	RF-006: <i>Carga de archivo de subtítulos</i> : Deberá estar cargado el archivo fuente
Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja
Fuente	Cliente

RF-008: Edición y traducción de subtítulos

Descripción	El usuario puede modificar los subtítulos individualmente. Además del texto dividido en dos líneas, puede: <ul style="list-style-type: none"> • Añadir anotaciones (para el revisor o el cliente). • Añadir marcas (para facilitar el acceso a subtítulos problemáticos o por revisar). • Poner el texto en cursiva o en negrita.
Prerrequisitos	RF-006: <i>Carga de archivo de subtítulos</i> o RF-007: <i>Creación de archivo de subtítulos</i> : Deberá haberse cargado algún archivo de subtítulos.
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Fuente	Cliente

RF-009: Indicación de longitud excesiva

Descripción	La aplicación debe indicar al usuario si los subtítulos superan el número máximo de caracteres por línea especificado. Igualmente, avisará si no se cumple la regla de los seis segundos.
Prerrequisitos	RF-006: <i>Carga de archivo de subtítulos</i> o RF-007: <i>Creación de archivo de subtítulos</i> : Deberá haberse cargado algún archivo de subtítulos.
Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja
Fuente	Cliente

RF-010: <i>Ir a subtítulo...</i>	
Descripción	El usuario tendrá la opción de ir a subtítulos determinados que puedan ser de especial interés, como: subtítulos marcados, subtítulos vacíos, subtítulos con observaciones, o número de subtítulo especificado.
Prerrequisitos	RF-006: <i>Carga de archivo de subtítulos</i> o RF-007: <i>Creación de archivo de subtítulos</i> : Deberá haberse cargado algún archivo de subtítulos.
Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja
Fuente	Cliente

RF-011: <i>Carga y visualización de subtítulos en pantalla de vídeo</i>	
Descripción	El usuario puede cargar el archivo de vídeo correspondiente a los subtítulos y visualizarlos sincronizados en pantalla.
Prerrequisitos	RF-006: <i>Carga de archivo de subtítulos</i> o RF-007: <i>Creación de archivo de subtítulos</i> : Deberá haberse cargado algún archivo de subtítulos.
Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja
Fuente	Cliente

RF-012: <i>Ocultación de pantalla de vídeo</i>	
Descripción	La pantalla de vídeo debe poder ocultarse o minimizarse cuando no se necesite.
Prerrequisitos	RF-011: <i>Carga y visualización de subtítulos en pantalla de vídeo</i>
Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja
Fuente	Cliente

RF-013: <i>Extracción automática de expresiones en subtítulos seleccionados</i>	
Descripción	El usuario debe poder extraer en un solo paso una lista de pares de subtítulos de los archivos fuente y destino, para su modificación e inclusión en el diccionario como pares de relaciones expresión - expresión. También debe poder especificar el ámbito, la categoría gramatical, el tipo de expresión, y añadir contexto y ejemplos si lo desea.
Prerrequisitos	RF-006: <i>Carga de archivo de subtítulos</i> , tanto fuente como destino.
Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja
Fuente	Cliente

RF-014: Visualización extendida o condensada de sugerencias

Descripción	En la ventana de subtitulación, las sugerencias deben poder condensarse y mostrar solo la información más básica, para poder mostrar un mayor número de entradas.		
Prerrequisitos			
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Fuente	Cliente		

RF-015: Consulta de estadísticas sobre el archivo de subtítulos

Descripción	El usuario puede consultar diversas estadísticas de estado del subtítulo actual, útiles para estimar el trabajo que queda por hacer. Dichas estadísticas incluyen número de subtítulos, de palabras, subtítulos marcados para revisar, subtítulos vacíos.		
Prerrequisitos	RF-006: <i>Carga de archivo de subtítulos</i>		
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Fuente	Cliente		

RF-016: Navegación y recolocación de ventanas

Descripción	El usuario puede navegar entre varias ventanas abiertas, ya sean de edición de diccionario, consulta de diccionario o edición de subtítulos, así como distribuir las ventanas para ver varias a la vez en pantalla.		
Prerrequisitos			
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Fuente	Cliente		

RF-017: Localización del sistema

Descripción	Puesto que se trata de una herramienta de traducción, el sistema debe ser localizable y permitir su empleo con los dos idiomas de trabajo.		
Prerrequisitos			
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Fuente	Cliente		

RF-018: Recordar preferencias del usuario

Descripción	El usuario puede modificar la configuración y el programa ha de recordar sus preferencias entre sesiones.		
Prerrequisitos			
Necesidad	<input type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input checked="" type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Fuente	Cliente		

RF-019: Impresión de entrada del diccionario

Descripción	El usuario puede imprimir una entrada del diccionario.		
Prerrequisitos	RF-001: <i>Consulta en diccionario</i>		
Necesidad	<input type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input checked="" type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input type="checkbox"/> Media	<input checked="" type="checkbox"/> Baja
Fuente	Cliente		

REQUISITOS FUNCIONALES DE RESTRICCIÓN
RFR-020: Modificación de idioma de entrada existente

Descripción	Para evitar inconsistencias con el resto de entradas del diccionario, la aplicación no permite modificar el idioma de una entrada ya existente.		
Prerrequisitos	RF-001: <i>Consulta en diccionario</i>		
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Fuente	Analista		

RFR-021: Modificación de códigos de tiempo

Descripción	Al no ser un programa de creación de subtítulos, y para mantener alineados en todo momento archivos fuente y destino para poder realizar las búsquedas en el diccionario, el sistema no permite modificar los códigos de tiempo.		
Prerrequisitos	RF-006: <i>Carga de archivo de subtítulos</i>		
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Fuente	Analista		

RFR-022: Varias ventanas de consulta de diccionario

Descripción	Para evitar la sobrecarga del sistema, y entendiendo que el usuario no necesita mantener abiertas varias ventanas de consulta, no se permite abrir más de una ventana de consulta del diccionario.		
Prerrequisitos			
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Fuente	Analista		

RFR-023: <i>Guardar entradas vacías</i>	
Descripción	No pueden guardarse entradas que tengan el campo de término o expresión vacío.
Prerrequisitos	
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja
Fuente	Analista

5.2.2 Requisitos no funcionales

Cómo debe funcionar el sistema y cómo se desarrollará.

RNF-024: <i>Construcción en plataforma .Net, con lenguaje C#</i>	
Descripción	El sistema se desarrollará sobre la plataforma .Net, empleando el lenguaje de programación C#.
Prerrequisitos	
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Fuente	Analista

RNF-025: <i>Instalación de Microsoft Visual C# Express o IDE similar</i>	
Descripción	El empleo del IDE simplifica las tareas más repetitivas, por lo que es recomendable su uso.
Prerrequisitos	
Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Fuente	Analista

RNF-026: <i>Uso de RDF para representar la información</i>	
Descripción	Se emplean expresiones RDF para describir la información en su almacenaje, y SPARQL para realizar consultas.
Prerrequisitos	
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Fuente	Analista

RNF-027: Empleo de la biblioteca SemWeb para C#/Net

Descripción	Se utiliza la biblioteca ya existente SemWeb.NET desarrollada por Joshua Tauberer, que permite el almacenamiento, la consulta y la recuperación de información mediante expresiones RDF.		
Prerrequisitos			
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Fuente	Analista		

RNF-028: Empleo de DirectX para reproducción de vídeo

Descripción	Se hará uso de la API de DirectX para reproducir el vídeo correspondiente a los subtítulos.		
Prerrequisitos			
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Fuente	Analista		

RNF-029: Creación de un control de usuario para mostrar los subtítulos

Descripción	Se opta por crear un control de usuario para mostrar cada subtítulo, asociado a los datos de éste. Los subtítulos fuente y destino se mostrarán alineados en dos columnas paralelas.		
Prerrequisitos			
Necesidad	<input type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input checked="" type="checkbox"/> Opcional
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Fuente	Analista		

RNF-030: Guardar cambios

Descripción	Hasta que el usuario no pulse en “Guardar cambios”, estos no se almacenarán, tanto en la ventana de subtítulos como en la de edición de diccionario.		
Prerrequisitos			
Necesidad	<input checked="" type="checkbox"/> Esencial	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Baja
Fuente	Analista		

RNF-031: Tiempo de consulta máximo: 3 segundos

Descripción	El tiempo máximo que debe tardar el sistema en realizar una consulta debe ser de tres segundos.		
Prerrequisitos			
Necesidad	<input type="checkbox"/> Esencial	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta	<input checked="" type="checkbox"/> Media	<input type="checkbox"/> Baja
Fuente	Analista		

RNF-032: <i>Extensibilidad a otros idiomas</i>	
Descripción	Aunque el sistema se crea para el par inglés-español, debe ser lo suficientemente flexible para poder emplearse con nuevos pares de idiomas sin requerir una reestructuración.
Prerrequisitos	
Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja
Fuente	Analista

RNF-033: <i>Procesos asíncronos</i>	
Descripción	La búsqueda en el diccionario se realizará desde hilos asíncronos, para permitir que el usuario pueda seguir empleando la aplicación mientras se realiza la consulta.
Prerrequisitos	
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja
Fuente	Analista

5.3 Especificación de casos de uso

Con los requisitos ya definidos, se puede pasar a describir los principales casos de uso del sistema, teniendo en cuenta que existe únicamente un tipo de usuario: el traductor.

Para describir los casos de uso, se empleará la siguiente plantilla:

Identificador: <i>Nombre</i>	
Descripción:	
Actores:	
Precondiciones:	
Pos condiciones:	
Excepciones:	
Necesidad:	<input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

Que contiene los siguientes campos:

- **Identificador:** Identifica unívocamente el caso de uso. El formato que tendrá es:

CU-XYX

Donde:

- X: especifica el escenario que recoge el caso de uso.
- Y: número que lo identifica dentro de una secuencia.
- **Nombre:** texto corto y descriptivo del caso de uso.
- **Descripción:** explica en pocas líneas en qué consiste el caso de uso.

- **Actores:** identifica quién realiza el caso de uso.
- **Precondiciones:** condiciones iniciales necesarias para que pueda darse el caso de uso.
- **Pos condiciones:** estado del sistema que se obtiene a continuación del caso de uso actual.
- **Excepciones:** cursos alternativos al desarrollo normal del caso de uso, en caso de suceder algún imprevisto.
- **Necesidad:** indica el nivel de obligatoriedad del caso de uso dentro de la aplicación final. Los valores posibles son: Esencial, Deseable u Opcional. Un caso de uso será Esencial si su existencia es imprescindible en el sistema; Deseable si es conveniente, pero no obligatorio, incluirlo y Opcional si su inclusión no es prioritaria.
- **Comentarios:** anotaciones relevantes al caso de uso en cuestión, como cursos alternativos o posibles modificaciones.

5.3.1 Escenario 1: Búsqueda de términos en el diccionario

Este escenario consiste en la consulta del diccionario por parte del traductor desde la ventana de consulta del diccionario. Puede buscar tanto sinónimos como traducciones, que la aplicación le mostrará debidamente.

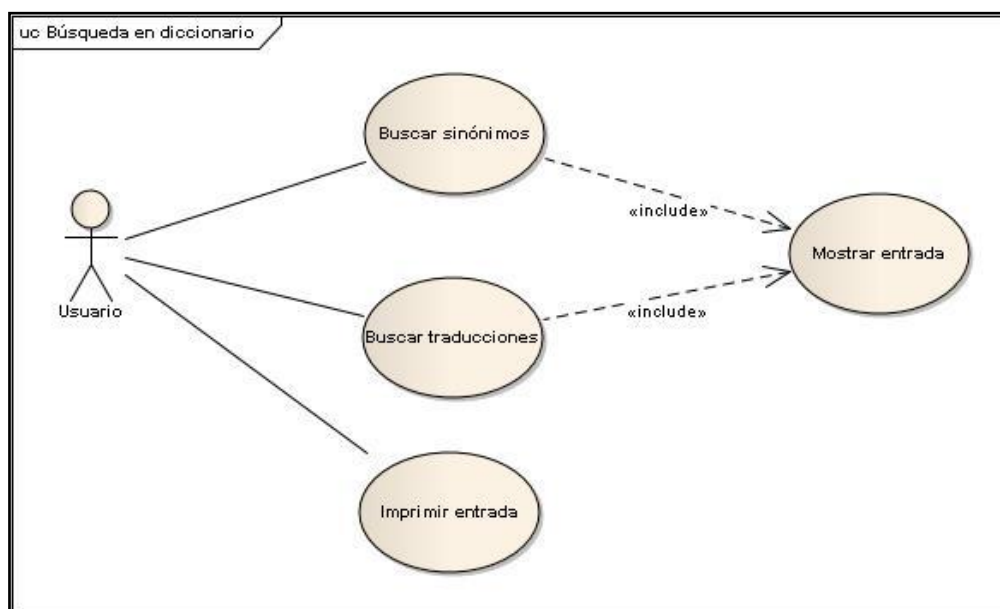


Ilustración 15: Diagrama de casos de uso del escenario 1

CU-101: Buscar sinónimos

Descripción:	El usuario introduce el término que desea buscar, especificando si se trata de un término en español o en inglés. La aplicación busca el término en el diccionario y muestra una lista de entradas que lo contienen. El usuario selecciona una entrada de la lista y la aplicación muestra los sinónimos correspondientes a dicha entrada.
Actores:	Usuario
Precondiciones:	
Pos condiciones:	CU-104: <i>Mostrar entrada</i> . Se muestra la entrada que coincide con los parámetros buscados por el usuario.
Excepciones:	
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

CU-102: Buscar traducción

Descripción:	El usuario introduce el término a buscar, especificando si busca una traducción de inglés a español, o viceversa. La aplicación busca el término en el diccionario y muestra una lista de entradas que lo contienen. El usuario selecciona una entrada de la lista y la aplicación muestra la traducción correspondiente a dicha entrada.
Actores:	Usuario
Precondiciones:	
Pos condiciones:	CU-104: <i>Mostrar entrada</i> . Se muestra la entrada que coincide con los parámetros buscados por el usuario.
Excepciones:	
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	Se pueden combinar CU-101 y CU-102 de modo que puedan buscarse tanto sinónimos como traducciones simultáneamente. El usuario incluye el término de búsqueda, y el sistema le muestra sinónimos, traducciones o ambos, según las preferencias de usuario.

CU-103: Imprimir entrada

Descripción:	El usuario especifica impresora y configuración de impresión e imprime la entrada (sinónimos o traducción) entregada en la última búsqueda.
Actores:	Usuario
Precondiciones:	CU-104: <i>Mostrar entrada</i> . Debe haberse mostrado una entrada al usuario tras haber efectuado una búsqueda.
Pos condiciones:	Se imprime la entrada en el formato especificado por el usuario.
Excepciones:	Si hay errores de impresión, se muestra un mensaje al usuario ofreciéndole la posibilidad de reintentar o cancelar la orden de impresión.
Necesidad:	<input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional
Comentarios:	

CU-104: <i>Mostrar entrada</i>	
Descripción:	La aplicación muestra la entrada correspondiente al término seleccionado por el usuario.
Actores:	Aplicación
Precondiciones:	CU-101: <i>Buscar sinónimos</i> o CU-102: <i>Buscar traducción</i> . El usuario debe haber realizado una búsqueda que haya tenido éxito y haya devuelto una entrada del diccionario.
Pos condiciones:	Se muestra la entrada que coincide con los parámetros buscados por el usuario.
Excepciones:	Si el término buscado no está recogido en el diccionario, se indicará la situación al usuario y se le permitirá buscar un nuevo término.
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

5.3.2 Escenario 2: Modificación del diccionario

El segundo escenario corresponde a la modificación del diccionario desde la ventana de edición, pudiendo editar los diversos campos de las entradas ya existentes, eliminar entradas del repositorio o introducir nuevas entradas.

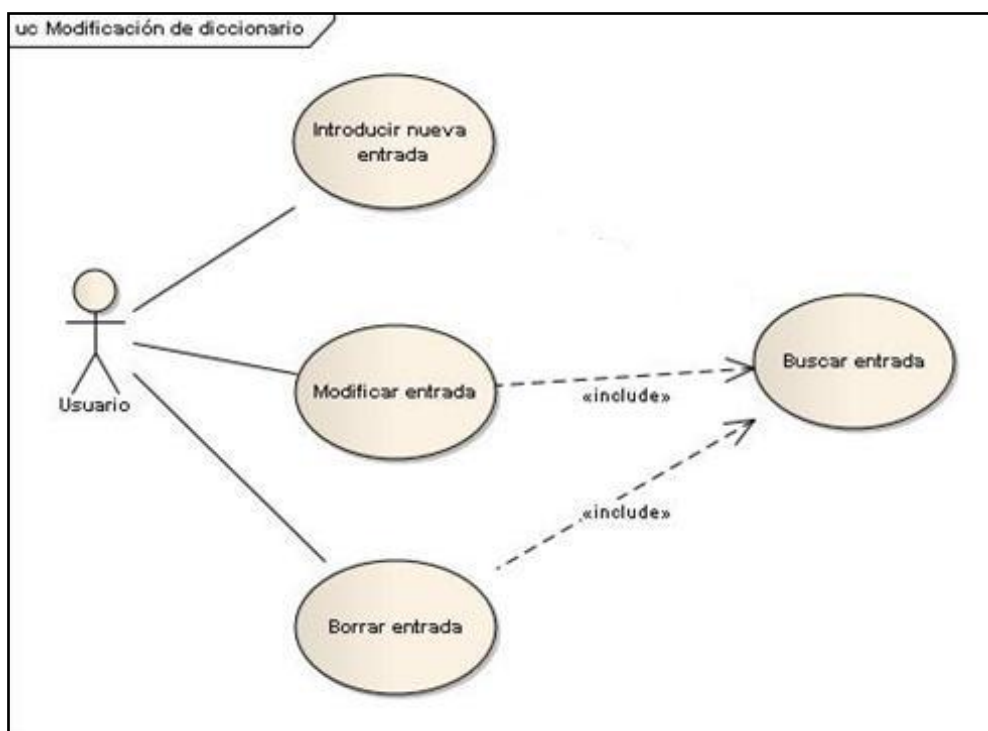


Ilustración 16: Diagrama de casos de uso del escenario 2

CU-201: <i>Buscar entrada</i>	
Descripción:	El sistema realiza la búsqueda de un término o expresión antes de modificarlo o borrarlo para comprobar que existe en el diccionario.
Actores:	Aplicación
Precondiciones:	
Pos condiciones:	CU-202: <i>Modificar término</i> o CU-204: <i>Borrar término</i> . Si está recogido en el diccionario, se permite su modificación o eliminación por parte del usuario.
Excepciones:	Si no existe ninguna correspondencia a la búsqueda, se informa al usuario y el sistema se mantiene a la espera de una nueva acción.
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

CU-202: <i>Introducir nueva entrada</i>	
Descripción:	El usuario escribe un nuevo término junto con su idioma y su función gramatical (nombre, verbo, etc.). La aplicación comprueba que no existe ya un recurso de esas características, y da al usuario la opción de añadir relaciones de sinonimia y traducción antes de almacenarlo en el repositorio.
Actores:	Usuario
Precondiciones:	
Pos condiciones:	CU-203: <i>Modificar entrada</i> . Se ofrece al usuario la posibilidad de añadir relaciones antes de almacenar el término.
Excepciones:	Si el usuario introduce un término ya existente en el diccionario, las relaciones repetidas no se introducirán de nuevo.
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

CU-203: *Modificar entrada*

Descripción:	El usuario busca un término en el diccionario, y la aplicación le ofrece una lista con las posibles correspondencias. El usuario selecciona una de dichas correspondencias, y la aplicación le muestra la entrada con todos los campos (idioma, categoría gramatical, tipo de expresión, grafía alternativa y comentarios), así como las relaciones de sinonimia y traducción que tiene la entrada. El usuario puede eliminar o modificar las relaciones existentes y añadir otras nuevas, además de añadir o editar comentarios, grafías alternativas y modificar la categoría gramatical y el tipo de expresión. Finalmente, se actualiza el diccionario con los cambios realizados.
Actores:	Usuario
Precondiciones:	CU-201: <i>Buscar entrada</i> o CU-204: <i>Introducir nueva entrada</i> . La entrada que se desea modificar debe existir en el diccionario, por lo que ha de realizarse una búsqueda previa; o bien debe ser una entrada que se va a introducir por primera vez en el diccionario.
Pos condiciones:	Se actualiza el diccionario con los cambios realizados por el usuario.
Excepciones:	
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	Se pueden modificar todos los campos de una entrada ya existente menos el idioma de la entrada, ya que esta puede formar parte de otras relaciones en el diccionario y podría dar lugar a incongruencias.

CU-204: *Borrar entrada*

Descripción:	El usuario busca un término en el diccionario, y el sistema le ofrece una lista con las posibles correspondencias. El usuario selecciona una de dichas correspondencias, y la aplicación le muestra la entrada, con los campos correspondientes y las relaciones de sinonimia y traducción. El usuario puede eliminar completamente la entrada seleccionada.
Actores:	Usuario
Precondiciones:	CU-201: <i>Buscar entrada</i> . La entrada que se desea eliminar debe existir en el diccionario, por lo que ha de realizarse una búsqueda previa.
Pos condiciones:	Se actualiza el diccionario con los cambios realizados por el usuario, eliminando la entrada.
Excepciones:	Si el término buscado por el usuario no está recogido en el diccionario, no se ofrece la opción de eliminarlo.
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	Si la entrada a eliminar es nueva, al eliminarla no se accede al diccionario, puesto que aún no se había guardado.

5.3.3 Escenario 3: Traducción de subtítulos

Este es el escenario general, ya que el traductor dedicará la mayor parte del tiempo de uso del sistema a la traducción, destinando un tiempo marginal a la introducción de términos en el diccionario.

En este escenario se contempla la carga de archivos de subtítulos, así como la búsqueda del texto en el diccionario para obtener sugerencias que poder aceptar o no. Igualmente, se puede abrir un formulario de edición de diccionario para la introducción de nuevos términos.

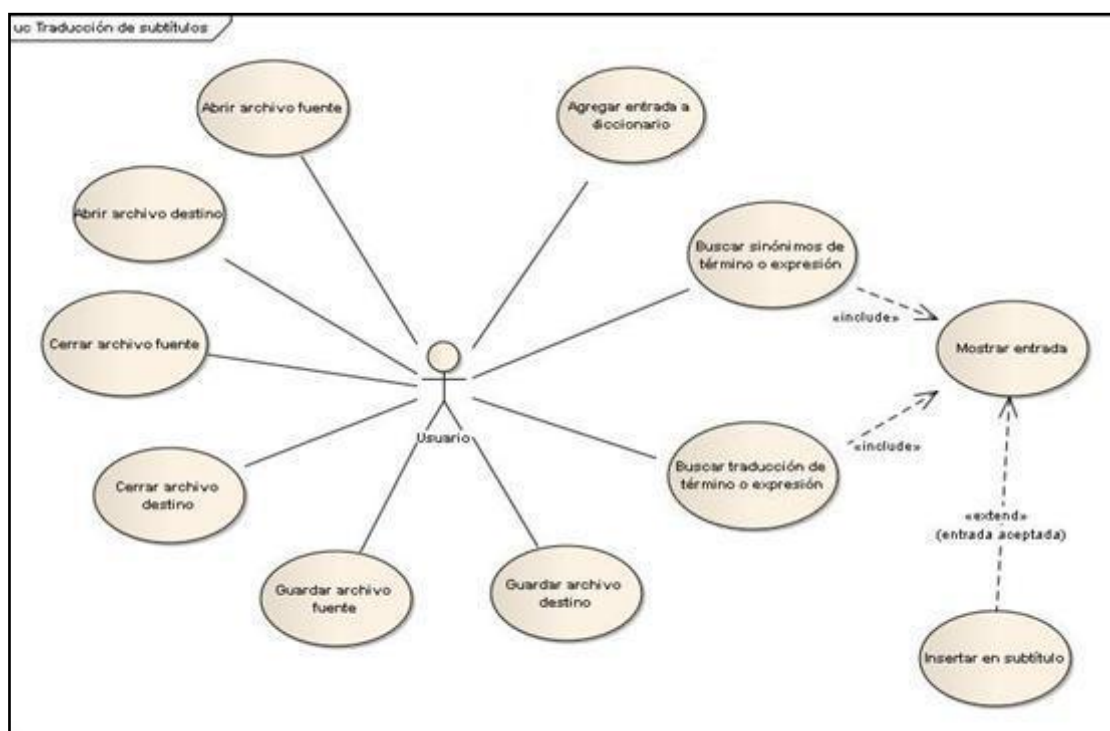


Ilustración 17: Diagrama de casos de uso del escenario 3

CU-301: Abrir archivo fuente	
Descripción:	El usuario abre el archivo de subtítulos en la lengua de origen (inglés).
Actores:	Usuario
Precondiciones:	
Pos condiciones:	Se abre el archivo de subtítulos en la lengua de origen en la columna correspondiente, para que el usuario proceda a su traducción.
Excepciones:	Si el archivo no existe o tiene un formato incompatible, se comunicará el error al usuario y se le ofrecerá la opción de abrir otro archivo o cancelar.
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

CU-302: Abrir archivo destino	
Descripción:	El usuario abre el archivo de subtítulos en la lengua meta (castellano).
Actores:	Usuario
Precondiciones:	
Pos condiciones:	Se abre el archivo de subtítulos en castellano en la columna correspondiente, para que el usuario proceda a la traducción del archivo fuente, o a la modificación del archivo destino.
Excepciones:	Si el archivo no existe o tiene un formato incompatible, se comunicará el error al usuario y se le ofrecerá la opción de abrir otro archivo o cancelar.
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

CU-303: Cerrar archivo fuente	
Descripción:	El usuario cierra el archivo de subtítulos en la lengua original (inglés). Si el archivo no ha sido guardado, se le indica al usuario y se le da la oportunidad de hacerlo antes de cerrar.
Actores:	Usuario
Precondiciones:	CU-301: <i>Abrir archivo fuente</i> . Para que se pueda cerrar el archivo, antes ha debido ser abierto por el usuario.
Pos condiciones:	Se cierra el archivo de subtítulos en la lengua de origen.
Excepciones:	
Necesidad:	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

CU-304: Cerrar archivo destino

Descripción:	El usuario cierra el archivo de subtítulos en la lengua meta (castellano). Si se han producido modificaciones después de que el usuario guardara el archivo por última vez, se le notificará y se le dará la posibilidad de guardarlo antes de cerrar.
Actores:	Usuario
Precondiciones:	CU-302: <i>Abrir archivo destino</i> . Para que se pueda cerrar el archivo, antes ha debido ser abierto por el usuario.
Pos condiciones:	Se cierra el archivo de subtítulos en la lengua meta.
Excepciones:	
Necesidad:	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

CU-305: Guardar archivo fuente

Descripción:	El usuario guarda el estado actual del archivo de subtítulos en la lengua original (inglés).
Actores:	Usuario
Precondiciones:	CU-301: <i>Abrir archivo fuente</i> . Para que se pueda guardar el archivo, antes ha debido ser abierto por el usuario.
Pos condiciones:	Se actualiza el archivo de subtítulos en inglés con los cambios realizados por el usuario.
Excepciones:	
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

CU-305: Guardar archivo destino

Descripción:	El usuario guarda el estado actual del archivo de subtítulos en la lengua meta (castellano).
Actores:	Usuario
Precondiciones:	CU-302: <i>Abrir archivo destino</i> . Para que se pueda guardar el archivo, antes ha debido ser abierto por el usuario.
Pos condiciones:	Se actualiza el archivo de subtítulos en castellano con los cambios realizados por el usuario.
Excepciones:	
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

CU-306: Buscar traducción de término

Descripción:	El usuario selecciona el término a buscar en la columna fuente y pulsa “Buscar”. Si se encuentra activada la opción “Traducción”, la aplicación buscará el término en el diccionario y mostrará todas las entradas que lo contienen, junto con la traducción correspondiente a cada una de las entradas.
Actores:	Usuario
Precondiciones:	CU-301: <i>Abrir archivo fuente</i> . El archivo en el idioma de partida debe haber sido abierto para que el usuario pueda seleccionar el término que desea traducir.
Pos condiciones:	CU-308: <i>Mostrar entrada</i> . Se muestran las entradas que coinciden con los parámetros buscados por el usuario.
Excepciones:	Si el término buscado no está recogido en el diccionario, se indicará la situación al usuario.
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	Si la opción de “Búsqueda automática” está activada, al entrar en el subtítulo se realiza automáticamente la búsqueda de traducciones de todo su contenido.

CU-307: Buscar sinónimos de término

Descripción:	El usuario selecciona el término a buscar, cuyo idioma detectará la aplicación en función de la columna en la que se encuentre dicho término. Tanto si es la columna fuente y se encuentra activada la opción “Corrección”, como si es la columna destino, la aplicación buscará el término en el diccionario y mostrará todas las entradas que lo contienen, junto con los sinónimos correspondientes a cada una de las entradas.
Actores:	Usuario
Precondiciones:	CU-301: <i>Abrir archivo fuente</i> o CU-302: <i>Abrir archivo destino</i> . Al menos uno de los dos archivos debe haber sido abierto para que el usuario pueda seleccionar el término que desea buscar.
Pos condiciones:	CU-308: <i>Mostrar entrada</i> . Se muestran las entradas que coinciden con los parámetros buscados por el usuario.
Excepciones:	Si el término buscado no está recogido en el diccionario, se indicará la situación al usuario.
Necesidad:	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	Si la opción de “Búsqueda automática” está activada, al entrar en el subtítulo se realiza automáticamente la búsqueda de sinónimos de todo su contenido.

CU-308: <i>Mostrar entrada</i>	
Descripción:	La aplicación muestra la entrada correspondiente al término buscado por el usuario.
Actores:	Aplicación
Precondiciones:	CU-306: <i>Buscar traducción de término</i> o CU-307: <i>Buscar sinónimos de término</i> . El usuario debe haber realizado una búsqueda que haya tenido éxito y haya devuelto una entrada del diccionario.
Pos condiciones:	Se muestra la entrada que coincide con los parámetros buscados por el usuario.
Excepciones:	Si no existe ninguna entrada que cumpla los criterios de búsqueda, se le indicará al usuario.
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

CU-309: <i>Insertar en destino</i>	
Descripción:	Si el usuario acepta alguna de las entradas proporcionadas por la aplicación tras una búsqueda, ésta se insertará en el subtítulo correspondiente de la columna que contiene la traducción.
Actores:	Usuario
Precondiciones:	CU-308: <i>Mostrar entrada</i> . La aplicación debe haber mostrado una entrada como respuesta a una búsqueda realizada por el usuario
Pos condiciones:	Se introduce el término en la columna destino.
Excepciones:	
Necesidad:	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	

CU-310: <i>Agregar entrada a diccionario</i>	
Descripción:	Si, tras realizar una búsqueda, la aplicación no ha proporcionado ninguna entrada satisfactoria al usuario, éste tiene la opción de añadir una nueva entrada al diccionario.
Actores:	Usuario
Precondiciones:	CU-301: <i>Abrir archivo fuente</i> o CU-302: <i>Abrir archivo destino</i> . Al menos uno de los dos archivos debe haber sido abierto para que el usuario pueda seleccionar el término con el que crear una nueva entrada.
Pos condiciones:	Se muestra un formulario de edición de término, donde el usuario podrá introducir los sinónimos, traducciones y todos los atributos pertinentes al nuevo término y añadirlo al diccionario.
Excepciones:	
Necesidad:	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional
Comentarios:	Como alternativa más rápida, el usuario tiene la opción de señalar todos los subtítulos que contengan texto que desee introducir en el diccionario, y realizar un procesamiento por lotes de los subtítulos seleccionados.

6. Diseño de la solución

6.1 Arquitectura del sistema

El sistema se diseñará mediante una arquitectura multicapa que facilite la escalabilidad y el mantenimiento de la aplicación. También permitirá modificar en el futuro algún componente sin afectar a los demás.

La arquitectura del sistema se muestra en la siguiente ilustración:

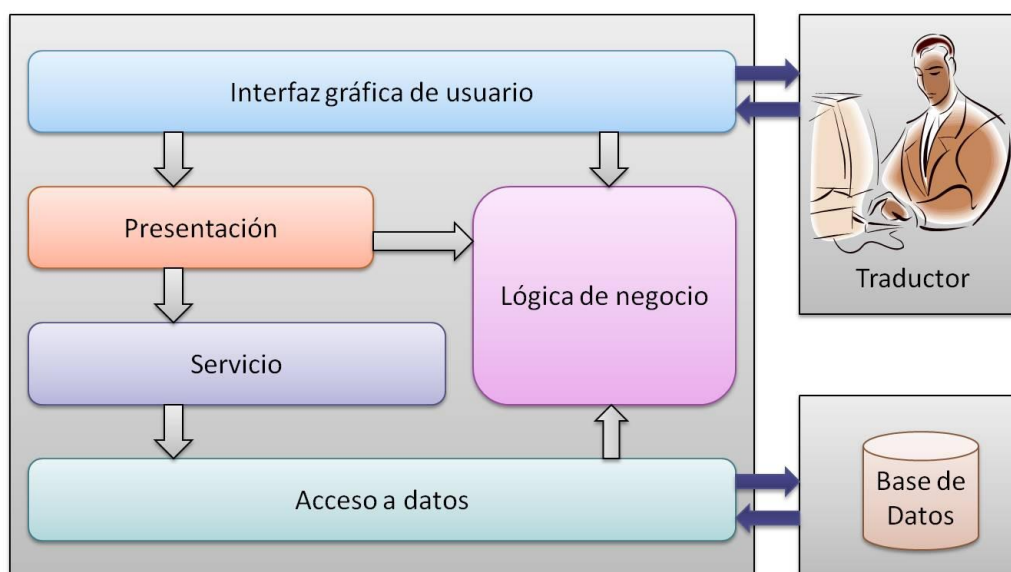


Ilustración 18: Diagrama de arquitectura

Las capas que componen el sistema son:

- **Interfaz gráfica de usuario (o Vista):** Formada por todos los formularios con los que interactuará el usuario, único punto de entrada de éste hacia el sistema. La lógica existente en esta capa se cede en lo posible a la capa de negocio y a la de presentación, pero siguen existiendo bastantes elementos específicos de la interfaz gráfica empleada (especialmente, en la ventana de subtitulación) que deberán gestionarse desde este componente. La interacción directa con la capa de negocio prácticamente se limita al enlace de los objetos, o en todo caso a la recepción y presentación de éstos, en varios formularios.
- **Presentación:** Contiene la lógica de presentación de datos más compleja, y aísla de la interfaz de usuario el acceso al repositorio. Se ha empleado un patrón de diseño “Presentador Supervisor” (40), variante del patrón Modelo-Vista-Presentador, que a su vez deriva del patrón Modelo-Vista-Controlador. La Vista desconoce su existencia, y la comunicación en la dirección Vista->Presentador se

produce mediante el lanzamiento de eventos a los que se suscribe el presentador. La capa de Presentación tiene a su vez un conocimiento limitado de la Vista, a través de una interfaz con una serie de eventos y propiedades. El resultado del procesamiento que haga esta capa con los objetos de negocios se devuelve a la Vista por medio de dichas propiedades, permaneciendo así opaca ante ésta.

- **Negocio o Modelo:** Contiene las clases de negocio del sistema, que se emplearán para intercambiar los datos entre la capa de acceso a datos y la vista. Desconoce la existencia de las demás capas.
- **Servicio:** Capa intermediaria entre la capa de Presentación y la de Acceso a Datos. Aísla las capas anteriores, de forma que si fuera necesario cambiar el sistema de almacenamiento de datos, únicamente habría que modificar o sustituir la capa correspondiente.
- **Acceso a datos:** Contiene las clases necesarias para convertir los datos al formato seleccionado y persistirlos.

6.2 Componentes del sistema

A continuación se profundiza en las capas presentadas anteriormente, describiendo los elementos o clases principales que componen cada una.

6.2.1 Interfaz gráfica de usuario

Es la encargada de interaccionar con el usuario y presentarle los datos correctamente, además de ofrecerle formularios para modificarlos y archivarlos. Se compone de un formulario principal MDI o pantalla contenedora, en el que podrán abrirse simultáneamente diversas ventanas hijas, tanto de consulta y edición del diccionario, como de edición de subtítulos. También contiene una serie de clases auxiliares y de formularios secundarios.

6.2.1.1 *Pantalla contenedora MDI (MDIParentForm)*

Es la pantalla inicial de la aplicación, y contendrá las pantallas hijas destinadas a acceder al diccionario y editar los subtítulos. Presentará una estructura compuesta de barra de menú, barra de herramientas, barra de pestañas y barra de estado.

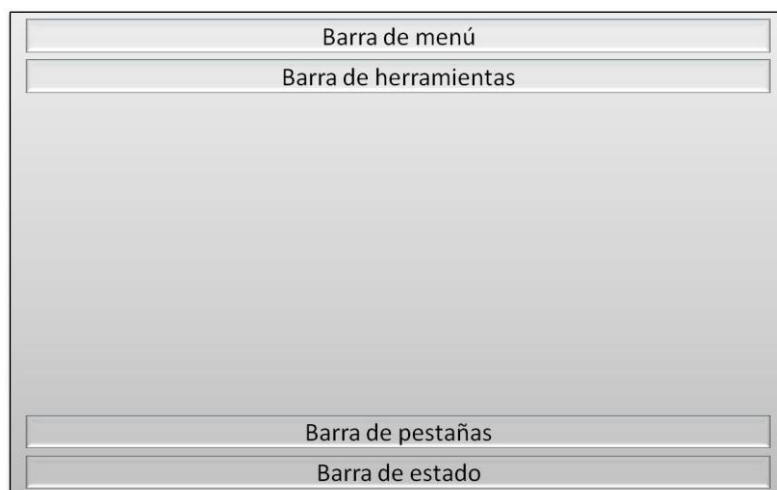


Ilustración 19: Pantalla contenedora MDI

Desde ella se crearán las ventanas hijas cuando el usuario pulse los elementos de menú o botones correspondientes.

6.2.1.2 *Pantalla de consulta del diccionario (DictionaryForm)*

Permitirá realizar consultas de términos o expresiones en el diccionario a través de una barra de búsqueda, mostrando un listado de entradas que se corresponden con la cadena de búsqueda, y a su derecha la entrada seleccionada en ese momento. El usuario podrá alternar entre todas las entradas del listado, y podrá también editar la entrada seleccionada, y validar las subentradas mostradas e incluirlas en el diccionario, en caso de que se haya realizado una búsqueda no segura.

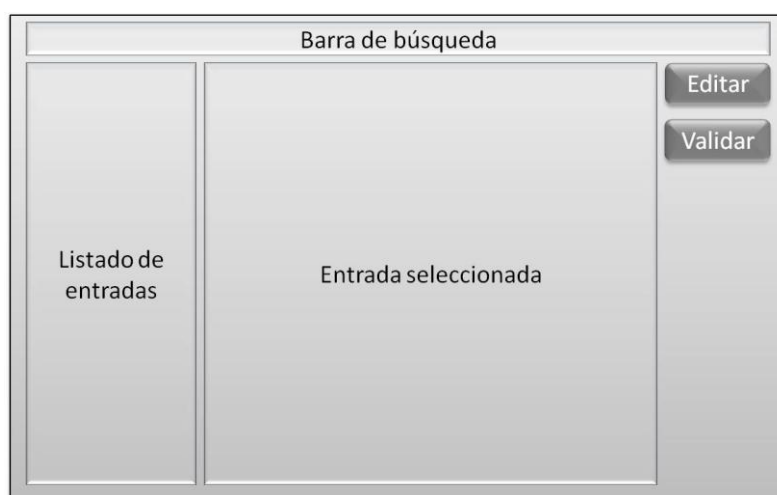


Ilustración 20: Pantalla de consulta del diccionario

6.2.1.3 *Pantalla de edición del diccionario (EditEntryForm)*

Tras realizar una búsqueda en el diccionario, muestra a la izquierda un listado de entradas halladas. La seleccionada por el usuario se muestra a la derecha del listado, pero en este caso en modo “edición”, con todos los campos, “entrada”, “grafía alternativa”, “comentarios”, “categoría gramatical” y “tipo de expresión” editables por el usuario.

En este mismo panel se incluyen dos tablas, una de traducciones y otra de sinónimos, cuyo contenido igualmente se puede modificar en su totalidad.

El usuario puede guardar la entrada creada o modificada, o cancelar los cambios.

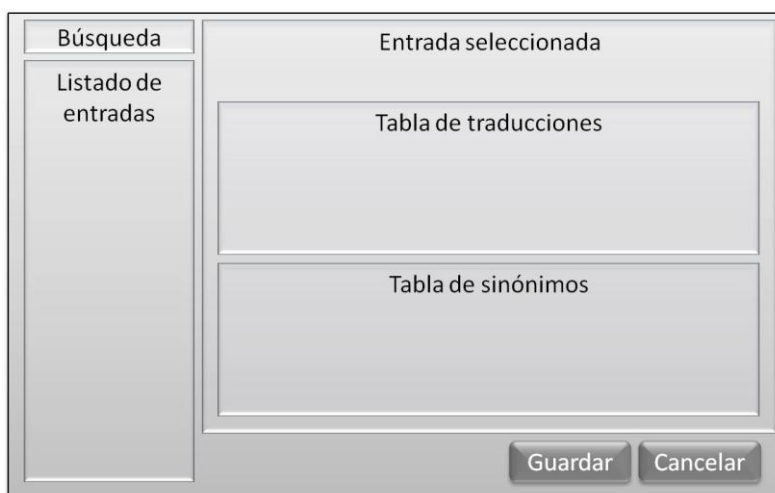


Ilustración 21: Pantalla de edición del diccionario

6.2.1.4 *Pantalla de traducción de subtítulos (SubtitleForm)*

En este tercer tipo de pantalla hija MDI se realizará la creación, edición y traducción de los subtítulos, con una columna para el archivo fuente, y otra para el archivo destino. A la derecha se muestran las sugerencias de traducción o sinonimia, según el modo en el que se esté trabajando (Traducción o Corrección). Un pequeño cuadro de búsqueda permite al usuario buscar términos o expresiones no presentes en los subtítulos.

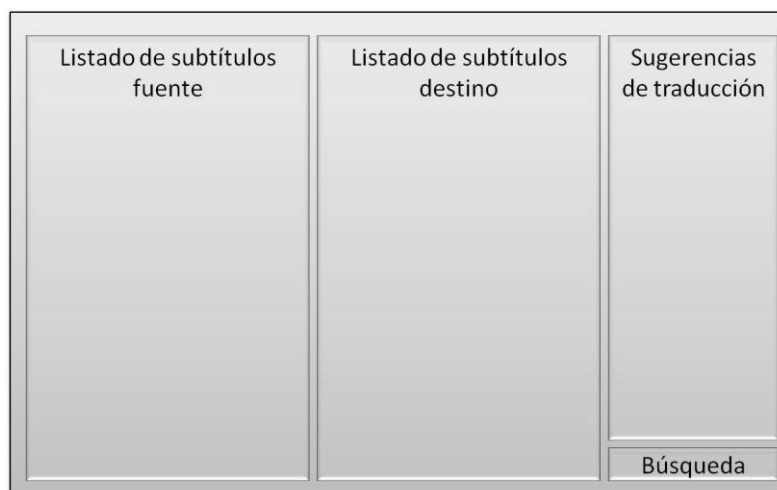


Ilustración 22: Pantalla de traducción de subtítulos

En esta pantalla también se puede cargar un archivo de vídeo para simular la reproducción de los subtítulos sobre la imagen. El vídeo puede ocultarse a voluntad, para maximizar la superficie destinada a los subtítulos cuando no se necesite ver la imagen.

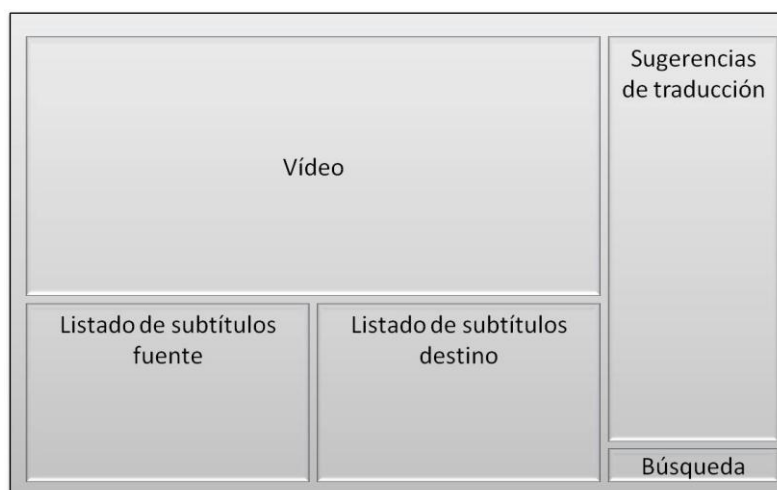


Ilustración 23: Pantalla de traducción de subtítulos con vídeo

6.2.1.5 *Otras clases y pantallas*

Esta capa contiene también una serie de clases auxiliares para facilitar la presentación de la información al usuario. Igualmente, se dispone de varios formularios secundarios, tales como el funcionario de Estadísticas, el de Opciones, el de Cambio de idioma y similares. Estos formularios presentarán una interfaz muy sencilla, por lo que no se considera necesaria su inclusión en este apartado.

6.2.2 Presentación

Tal como se mencionó en el apartado de arquitectura, en este sistema se emplea un componente de presentación que se encarga de la lógica más avanzada de presentación de datos, aligerando así la capa de interfaz gráfica de usuario. Desacoplando en parte el Modelo de Negocio de la Vista, se podrán realizar modificaciones de esta (incluso cambiarla por completo) que no afecten al resto del sistema.

La creación y almacenamiento de datos, el mantenimiento de un historial de últimos elementos visitados o consultados, la apertura de archivos, la adquisición del elemento a mostrar al usuario, entre otras funciones, se gestionan desde esta capa.

Existe un presentador para cada una de las tres pantallas MDI hijas: presentador de consulta de diccionario (DictionaryPresenter), presentador de edición de diccionario (EditEntryPresenter) y presentador de traducción de subtítulos (SubtitlePresenter).

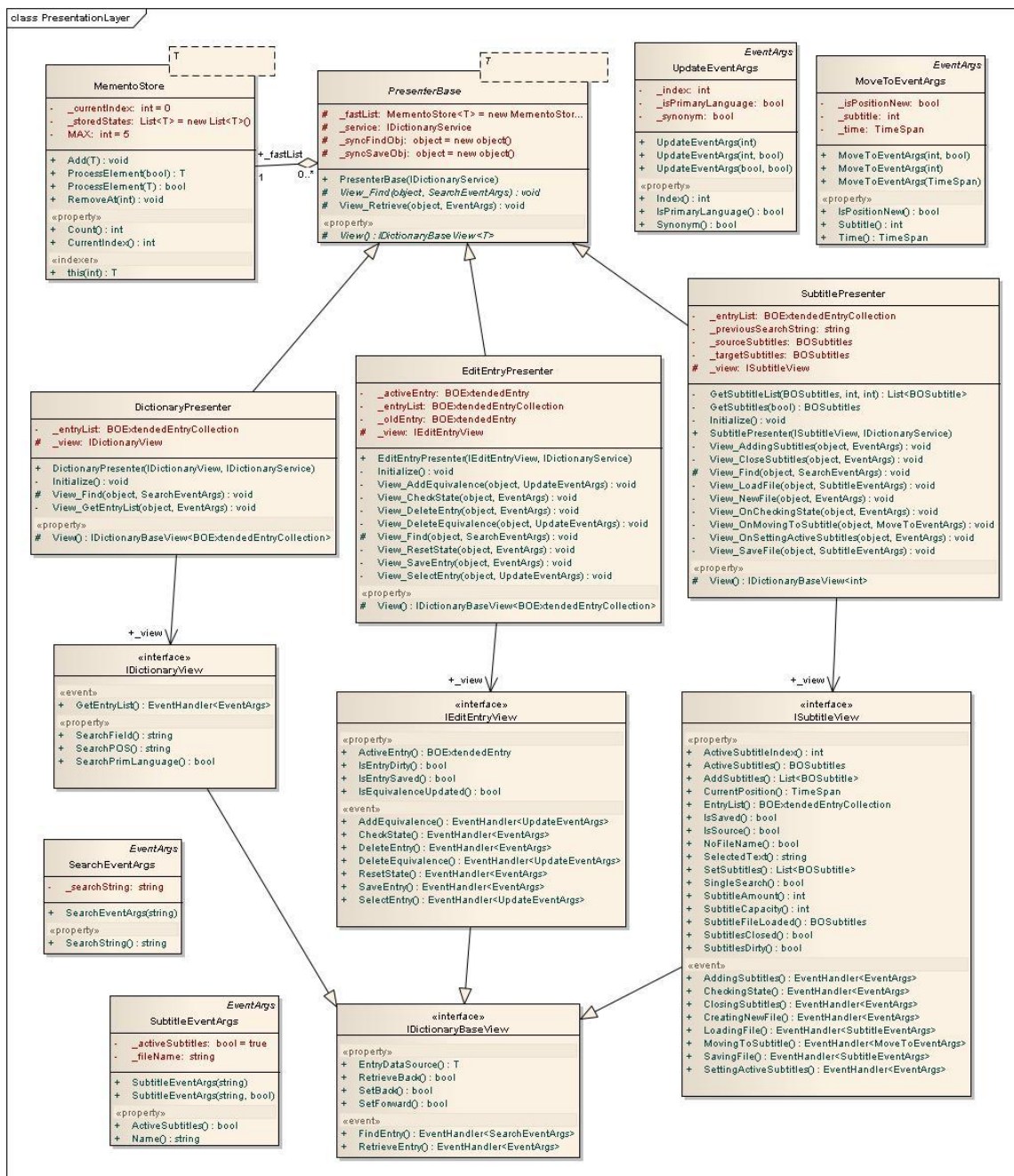


Ilustración 24: Diagrama de clases del componente Presentación

Como se observa en el diagrama de clases, los tres presentadores heredan de una clase base de presentación, y cada uno añade sus propios métodos y propiedades. Igualmente, los tres presentadores exigen una serie de eventos y propiedades a sus respectivas vistas, a través de interfaces que mantienen el resto de la vista opaca al presentador.

Se emplearán también una serie de clases de argumentos de eventos, que heredarán de la clase EventArgs, y para mantener un historial de los últimos subtítulos visitados (en el caso de la vista de subtitulación) o de las últimas consultas realizadas

(en el caso de las vistas de consulta y edición de diccionario) se implementará un patrón de diseño Memento.

6.2.3 Negocio

La capa de negocio contiene las clases de negocios utilizadas para modelar las entidades representadas por el sistema. Al agrupar las propiedades y la lógica correspondientes a cada una, se consigue abstraer su contenido y lograr un manejo más intuitivo que si se trabajara directamente con funciones y variables simples. Las clases más relevantes son:

- **Subtítulo (BOSubtitle):** almacena los datos que componen cada subtítulo, como: primera y segunda líneas del subtítulo, códigos de tiempo de entrada y de salida, caracteres en cursiva o negrita, anotaciones y marcas. También contiene un método estático que permite la creación de un nuevo subtítulo, y un método para actualizarlo con las modificaciones del usuario.
- **Colección de subtítulos (BOSubtitles):** contiene ordenados todos los subtítulos importados de un archivo determinado, así como métodos para cargar, guardar y procesar los subtítulos.
- **Entrada (BOEntry):** contiene los campos básicos de la entrada creada o buscada: grafía, idioma, tipo de expresión, categoría gramatical, grafía alternativa y anotaciones.
- **Equivalencia semántica (BOSEmanticEquivalence):** adecuada tanto para contener un sinónimo como una traducción, contiene la entrada BOEntry de esa equivalencia, además de una serie de elementos asociados a la relación de equivalencia, tales como ámbito de aplicación, contexto, ejemplos y, en caso de haberse obtenido mediante búsqueda no segura, la cadena de vínculos en el grafo que enlaza la entrada original con la equivalencia.
- **Colección de equivalencias (BOSEmanticEquivalenceList):** lista de varias equivalencias semánticas asociadas a una entrada.
- **Entrada extendida (BOExtendedEntry):** contiene la entrada creada o recuperada del repositorio BOEntry, con los campos específicos de esa entrada única, así como dos colecciones de equivalencias semánticas de esa entrada, una para traducciones y otra para sinónimos.
- **Colección de entradas extendidas (BOExtendedEntryCollection):** lista con todas las entradas obtenidas en la consulta del diccionario, o todas las entradas a insertar en éste tras el procesamiento por lotes de los subtítulos.

- **Otras clases:** Además de las anteriores, en esta capa hay una serie de clases auxiliares, de entre las cuales caben destacar dos:
 - **Validación (Validation):** Las clases que heredan de BOBase contienen una instancia de esta clase, que se encarga de comprobar que los valores introducidos en los distintos campos son adecuados, y en caso contrario, se genera un mensaje de error que se puede comunicar al usuario.
 - **Criterios (Criteria):** contiene los criterios de búsqueda a tener en cuenta durante la consulta del diccionario, tales como idioma de partida y de llegada de la búsqueda, número máximo de resultados, si se trata de una búsqueda segura, etcétera. Se pasa como argumento al realizar la consulta.

El diagrama de clases completo de esta capa se muestra en la ilustración 27, mientras que en las ilustraciones 25 y 26 se muestran en más detalle las clases principales.

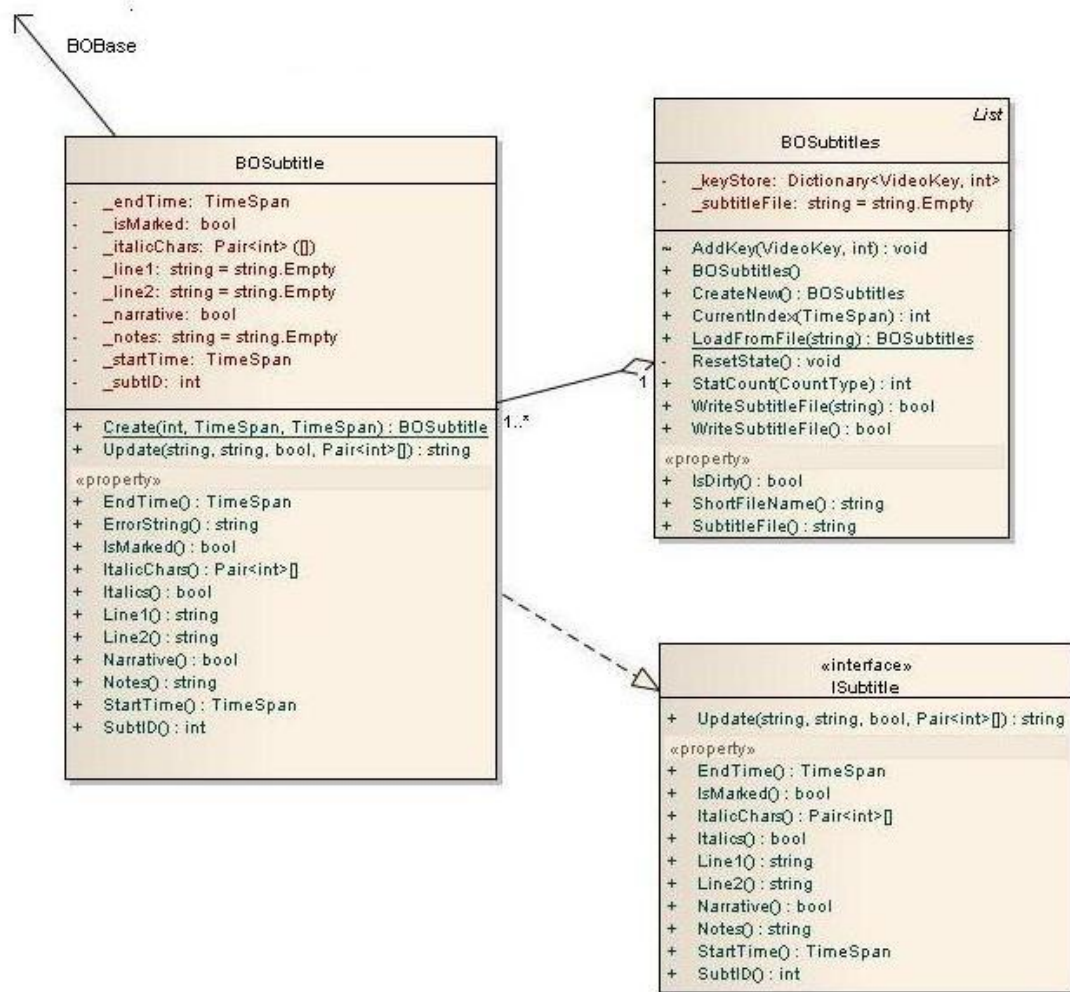


Ilustración 25: Diagrama de clases – Negocios: subtítulos

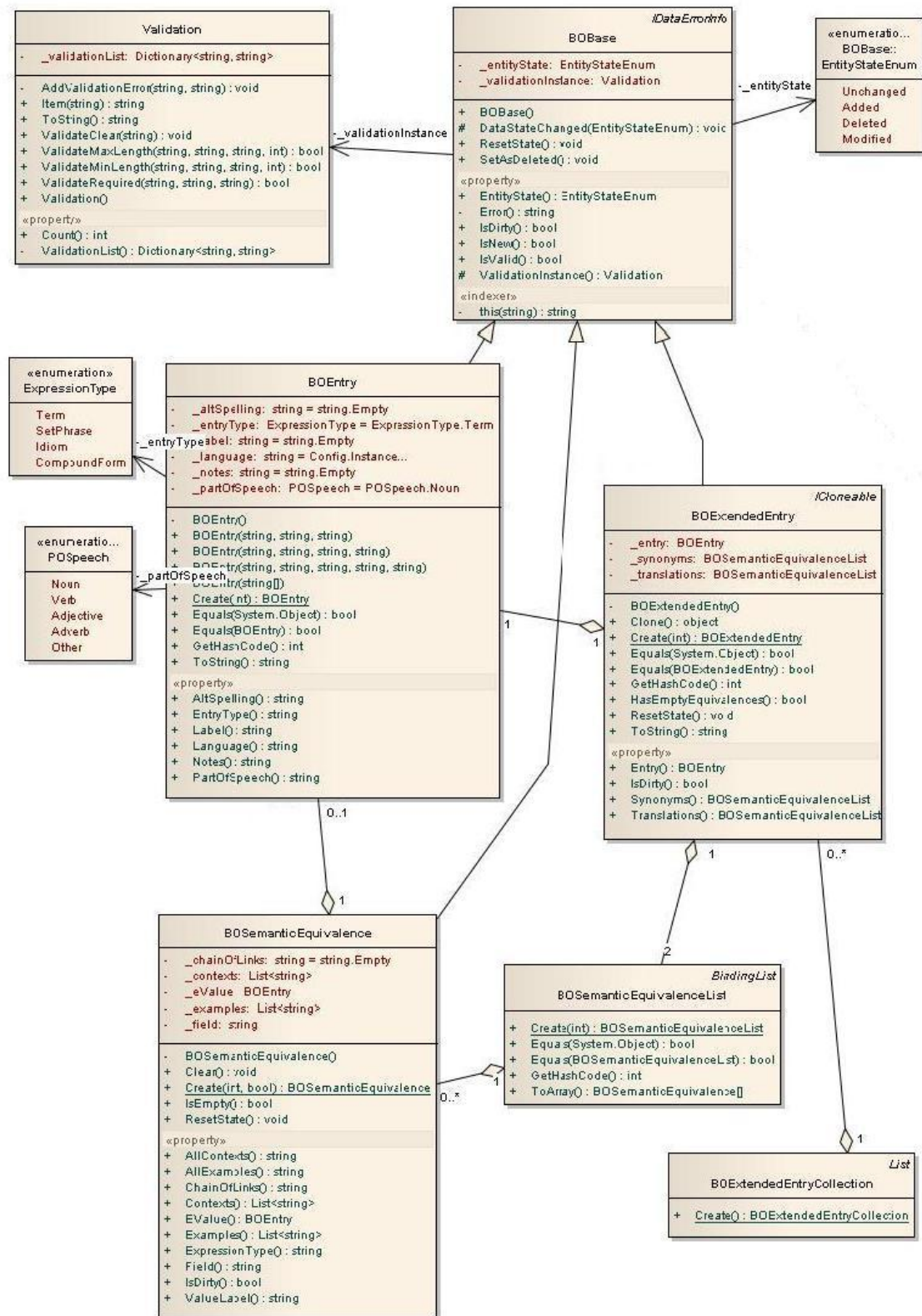


Ilustración 26: Diagrama de clases - Negocios: entradas, validación y BOBase

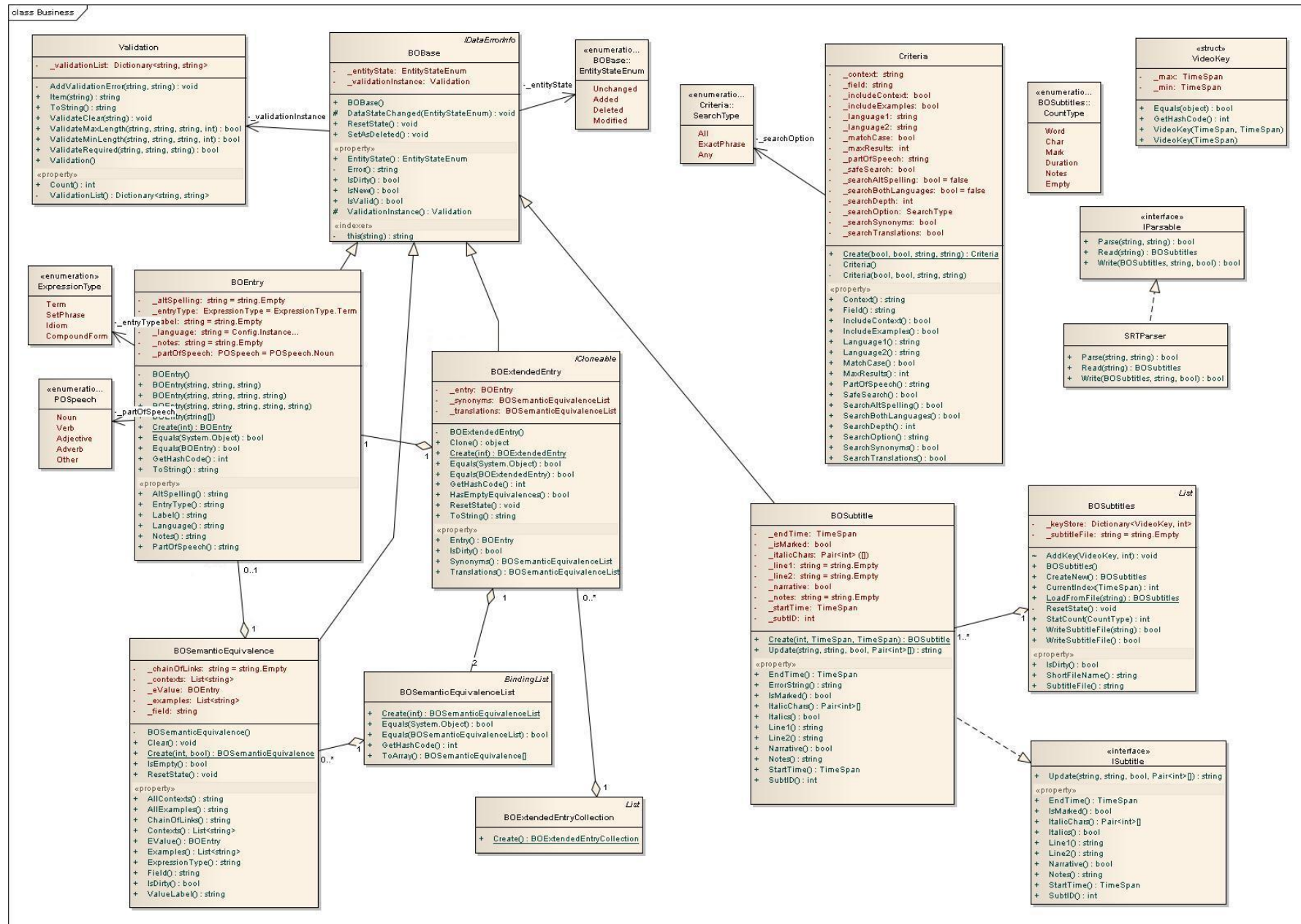


Ilustración 27: Diagrama de clases de del componente Negocio

6.2.4 Servicio

Esta capa consta de una única clase que enlaza la capa de presentación con la de acceso a datos, por medio de varios métodos de acceso. Se empleará un patrón de diseño Singleton, para restringir la instanciación a un único objeto.

El uso de esta capa se justifica porque desacopla el resto del sistema del método de almacenamiento y persistencia de los datos, de modo que si en un futuro decide utilizarse otra tecnología o modificar la modelización de la información utilizada, bastará con cambiar la capa de acceso a datos y el resto del sistema podrá seguir inalterado.

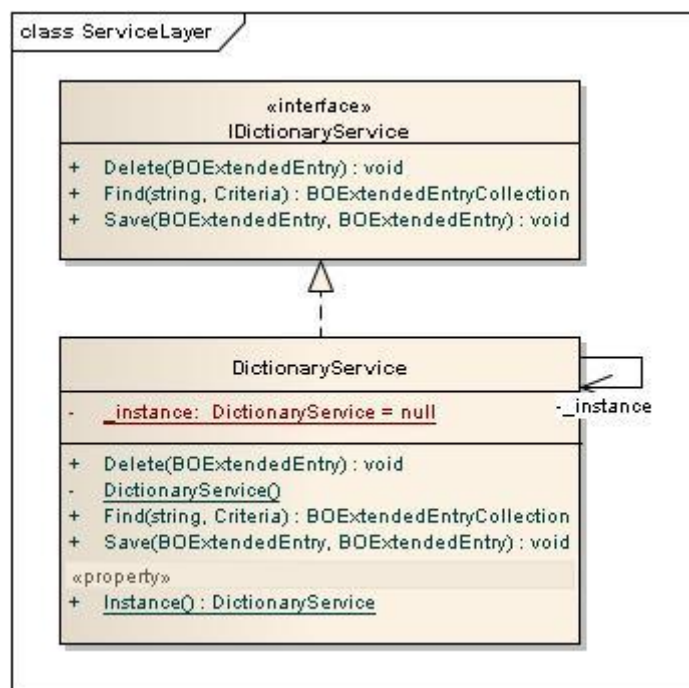


Ilustración 28: Diagrama de clases del componente Servicio

6.2.5 Acceso a datos

Esta capa consta de dos clases: la clase de acceso a datos, `DALEntry`, y la clase de sumidero de datos, `EntryQuerySink`, que almacenará los resultados de la consulta SPARQL al repositorio. Además, la estructura `DALConstants` proporciona a ambas clases varias cadenas de texto que éstas emplean durante la comunicación con el repositorio y la conversión de las URI.

La clase `DALEntry` contiene métodos para la conexión con el diccionario para buscar o introducir expresiones, y métodos para la conversión de los objetos de negocios a tripletas almacenables en un repositorio RDF.

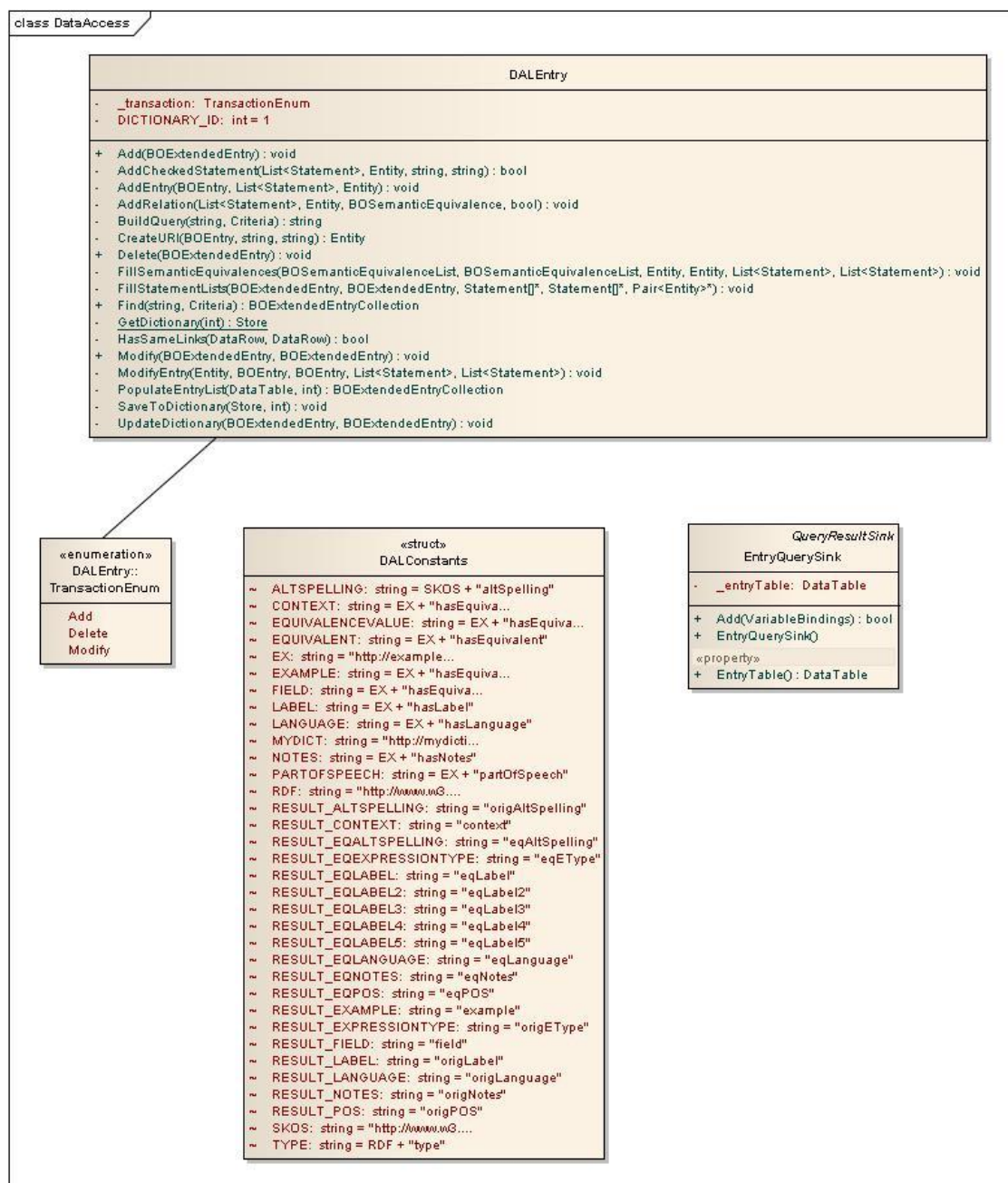


Ilustración 29: Diagrama de clases del componente Acceso a datos

6.3 Diagrama de paquetes

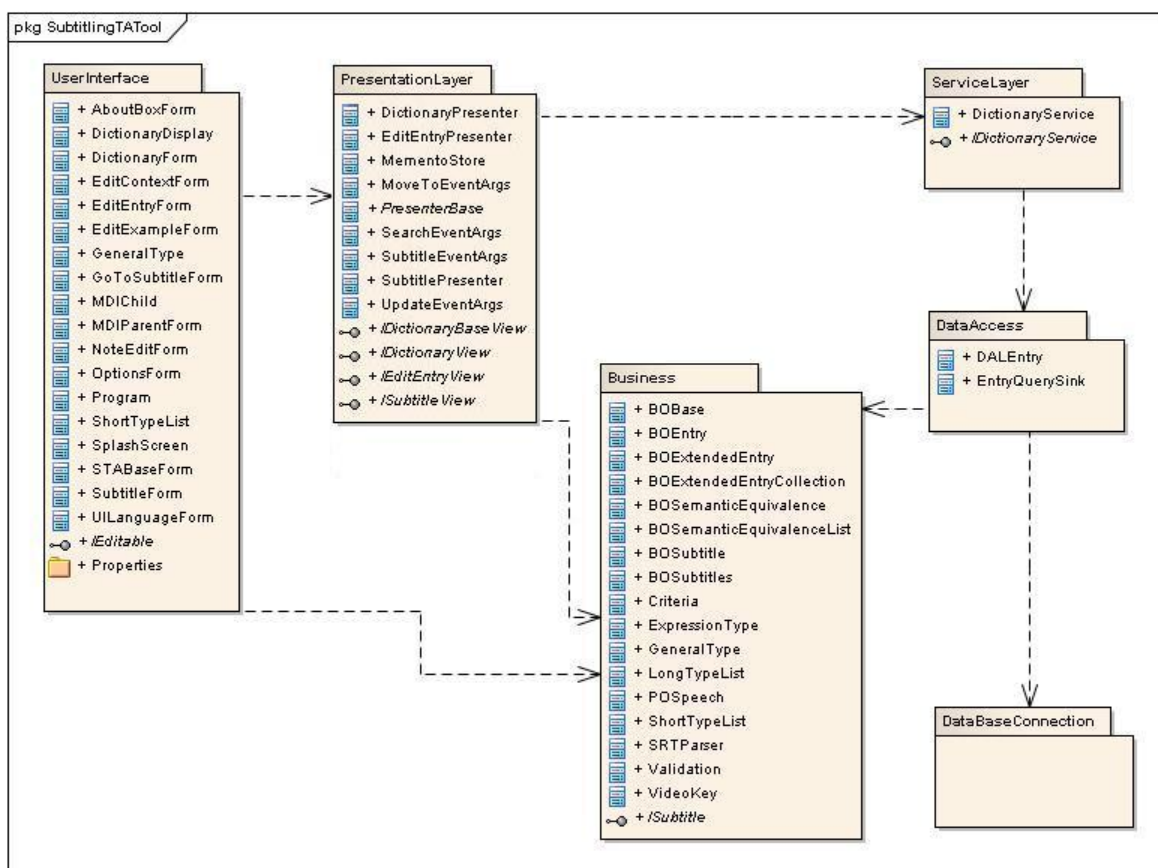


Ilustración 30: Diagrama de paquetes del sistema

6.4 Repositorio terminológico

Tal como se comentó en la introducción, la información se modelará mediante una ontología RDF, por lo que las entradas y todos sus campos y relaciones habrán de descomponerse en tripletas. Teniendo esto en cuenta, así como los principios de diseño descritos en apartados anteriores, se diseñará una ontología adecuada para el dominio de la aplicación, empleando la metodología de desarrollo ya expuesta.

A la hora de diseñar la ontología, se han tenido en cuenta diversas consideraciones:

- Un factor clave es el hecho de que no se va a tratar de una introducción sistemática de entradas y acepciones claramente estructuradas, sino que el usuario de la aplicación (traductor) será quien cree el repositorio sobre la marcha. La flexibilidad que esto requiere implica que la entrada a almacenar será la representación gráfica del concepto, y no el concepto en sí. Así, para introducir nuevas entradas, el traductor no tendrá que comprobar previamente si la misma entrada y todos los sinónimos o traducciones que desea añadir se encuentran ya

presentes en otras relaciones y seleccionar los más adecuados según su acepción, lo que supondría una tarea excesivamente tediosa.

- Si bien en un principio se pensó en crear las clases “Expresión en Inglés” y “Expresión en Español” y los predicados “Traducción” y “Sinónimo”, pronto se descartó esta idea a favor de una representación más flexible y ampliable. Así, a la clase única “Expresión” se le añade una propiedad “Idioma”, y los predicados de traducción y sinonimia se aúnan en uno solo, “Equivalencia Semántica”. Las respectivas propiedades “Idioma” de cada expresión indicarán si se trata de una traducción entre dos lenguas o de una equivalencia monolingüe.
- Conviene emplear la lengua inglesa para definir los elementos de la ontología creada, para facilitar su reutilización.
- Se utilizarán predicados y clases de otras ontologías aplicables a este contexto. Además de RDF y RDFS, se emplearán elementos de:
 - Dublin Core Metadata Element Set (DC) (41): vocabulario de propiedades para describir recursos, tales como Creador, Descripción, Fuente o Idioma.
 - Simple Knowledge Organization System (SKOS) (41): modelo de datos para compartir y enlazar sistemas de organización de datos, como los tesauros o taxonomías, vía Internet.
 - The Suggested Upper Merged Ontology (SUMO) (43): ontología general creada con el objetivo de facilitar el intercambio de datos, la búsqueda de información, el razonamiento automático y el procesamiento del lenguaje natural.
- Debido precisamente a la sencillez que lo caracteriza, RDF presenta una limitación cuando se trata de establecer relaciones n-arias. En el repositorio se desean indicar algunos atributos, como el ámbito o los contextos, para una determinada relación de equivalencia semántica, pero la estructura de RDF no permite añadir atributos a la relación en sí, tal como se observa en la ilustración 31. La solución que se empleará en esta ontología será la propuesta por el W3C en (44): representar la relación no como una propiedad, sino como una clase, en este caso “Relación de Equivalencia”. Las instancias individuales de esta clase corresponderán a las instancias de las distintas relaciones de equivalencia, y podrán aplicársele tantas propiedades como se desee.

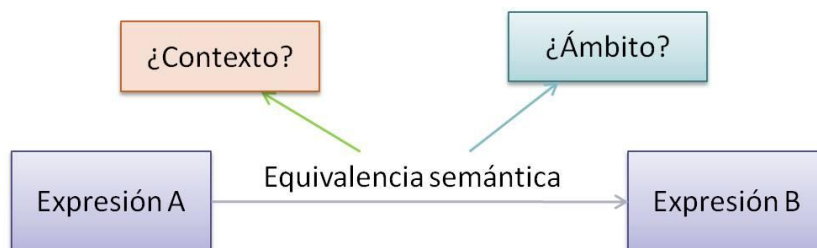


Ilustración 31: Relación N-aria no posible en RDF

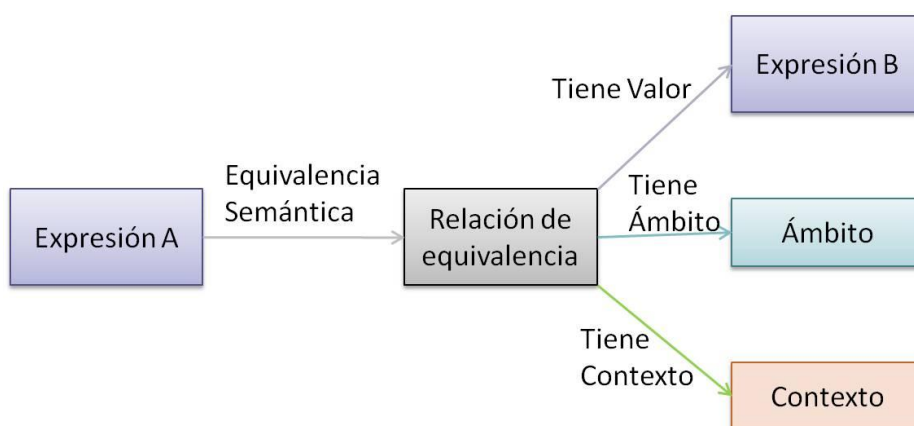


Ilustración 32: Relación N-aria en RDF con nodo de relación

El grafo de la ontología RDF creada a partir de las consideraciones anteriores se presenta en la ilustración 33.

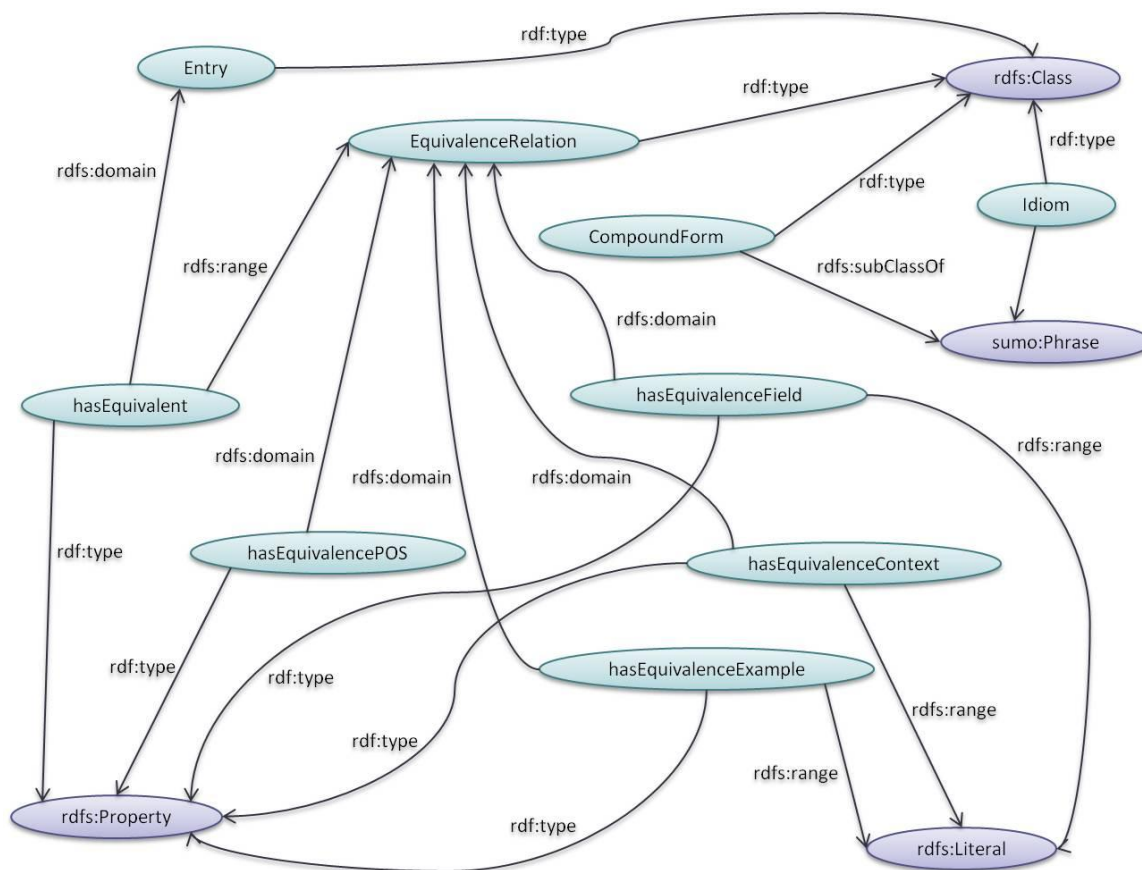


Ilustración 33: Ontología RDF creada

Además de los elementos definidos aquí, se hará uso de varias siguientes propiedades y clases RDF existentes en otras ontologías, que se recogen en la tabla 11.

ELEMENTO	TIPO	ONTOLOGÍA	DESCRIPCIÓN
language	Propiedad	DC	Idioma del recurso.
altSpelling	Propiedad	SKOS	Grafía alternativa para el recurso.
note	Propiedad	SKOS	Comentarios o anotaciones sobre el recurso.
Noun	Clase	SUMO	Categoría gramatical de los sustantivos.
Verb	Clase	SUMO	Categoría gramatical de los verbos.
Adjective	Clase	SUMO	Categoría gramatical de los adjetivos.
Adverb	Clase	SUMO	Categoría gramatical de los adverbios.
Word	Clase	SUMO	El recurso lingüístico al que se refiere es una palabra unitaria.
SetPhrase	Clase	SUMO	El recurso lingüístico al que se refiere es una locución (dos o más palabras).
label	Propiedad	RDFS	Representación gráfica del recurso.
type	Propiedad	RDF	Tipo de recurso (en este caso, el tipo de expresión: palabra, locución, modismo o forma compuesta).
Value	Propiedad	RDF	Valor del recurso (en este caso, de la relación de equivalencia. Es el recurso equivalente – traducción o sinónimo- al sujeto de dicha relación).

Tabla 11: Elementos de ontologías existentes empleados

6.5 Diseño de la base de datos

El diagrama entidad-relación del sistema de almacenamiento se muestra en la ilustración 34.

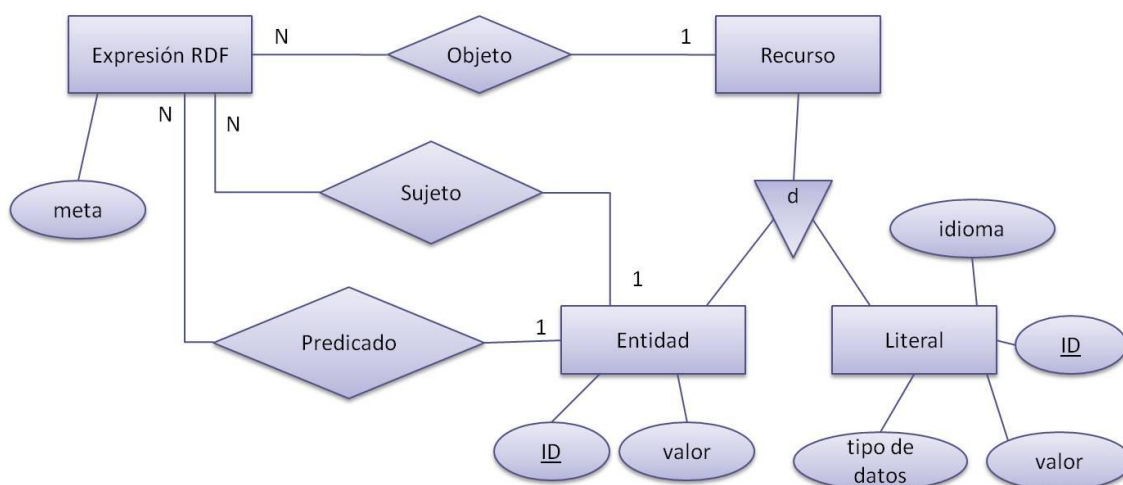


Ilustración 34: Diagrama Entidad-Relación del sistema

La estructura de la base de datos viene determinada por la librería SemWeb.NET (38) que se utilizará para manipular y almacenar las tripletas RDF. Dicha librería permite el uso de almacenes de persistencia MySQL, PostgreSQL y Sqlite, además de archivos de texto.

En los almacenes SQL se almacenan los datos en tres tablas. La principal es `rdf_statements`, que contiene una fila por cada expresión o tripleta que hay en el almacén. Las columnas son identificadores numéricos correspondientes a sujeto, predicado, objeto y metadatos. Los identificadores se enlazan con URIref, para las entidades, y valores, para los literales, en las tablas `rdf_entities` y `rdf_literals`, respectivamente.

7. Implementación

Una vez realizados el análisis y el diseño de la solución, puede procederse a su desarrollo. La tecnología empleada será la plataforma .Net, y para automatizar una parte de la generación del código, además de simplificar el diseño de la interfaz gráfica de usuario, se utilizará el entorno integrado de programación Visual Studio C# Express (45), de Microsoft.

7.1 Plataforma de desarrollo .Net

La plataforma .Net (45) se compone de una serie de herramientas y librerías que favorecen el desarrollo de software. Algunas de sus características son:

- Proporciona un entorno de programación orientada a objetos coherente y flexible.
- El entorno de ejecución de código y la Biblioteca de Clases reducen el tiempo de implementación a la vez que facilitan el empleo de diversos elementos como las conexiones con bases de datos, la lectura y escritura de archivos o el diseño de interfaces de usuario.
- Ofrece un entorno coherente para desarrollar distintos tipos de herramientas, tales como las aplicaciones de escritorio o las aplicaciones web.

El principal componente de la plataforma, el Common Language Runtime (CLR), es un motor que administra el código en tiempo de ejecución y proporciona servicios centrales como la administración de memoria, la ejecución de subprocesos, la ejecución de código o la comprobación de la seguridad del código.

El lenguaje de programación que se utilizará es C#, lenguaje orientado a objetos diseñado para crear aplicaciones ejecutadas sobre la plataforma .Net.

7.2 Codificación

Puesto que en apartados anteriores ya se ha descrito el diseño de cada componente, este apartado se centrará en los aspectos más relevantes de la implementación de cada uno. Debido a la extensión del código, se seleccionarán únicamente aquellos fragmentos que presentan mayor importancia o interés.

7.2.1 Interfaz gráfica de usuario

La interfaz gráfica de usuario (IGU) se compone de una serie de ventanas con las que interactuará el traductor. El código necesario para la creación e inicialización de los componentes gráficos ha corrido a cargo del entorno de programación, por lo que no se incluirá en este documento.

La ventana principal es el formulario contenedor MDI, MDIParentForm. Desde esta clase se crearán las ventanas hijas, se gestionará la alternancia entre ellas mediante las pestañas y se realizará la comunicación con ellas cuando el usuario seleccione elementos de menú o pulse botones de la barra de herramientas. La gestión de los eventos de los controles se realiza mediante la suscripción a éstos de métodos gestores de eventos, tales como el presentado a continuación.

```
// Comunica a la ventana actual, si es de subtítulo, que debe ir
// al siguiente subtítulo del tipo que se le indica
// El tipo se indica mediante un elemento de la enumeración GoToType
private void GoTo(object sender, EventArgs e)
{
    if (this.ActiveMdiChild is SubtitleLabel)
    {
        if (sender == uxMarker)
            (this.ActiveMdiChild as SubtitleLabel).GoTo(GoToType.Marker);
        else if (sender == uxNotes)
            (this.ActiveMdiChild as SubtitleLabel).GoTo(GoToType.Notes);
        else if (sender == uxEmptySubtitle)
            (this.ActiveMdiChild as SubtitleLabel).GoTo(GoToType.EmptySubtitle);
        else if ((sender == uxGoToSubtitle) || (sender == uxTSGoToSubtitle))
            (this.ActiveMdiChild as SubtitleLabel).ChangeSubtitle();
    }
}
```

La navegación entre ventanas MDI hijas se realiza por medio de pestañas que se muestran en una barra en el contenedor MDI. El sistema empleado para asociar una pestaña con una ventana hija ha sido utilizar la propiedad Tag del formulario y asociarla a su pestaña correspondiente, y viceversa. A continuación se muestra parte de dicha implementación.

```
private void uxWindowTabs_SelectedIndexChanged(object sender, EventArgs e)
{
    if ((uxWindowTabs.SelectedTab != null)
        && (uxWindowTabs.SelectedTab.Tag != null)) {
        (uxWindowTabs.SelectedTab.Tag as Form).Select();
    }
}

private void MDIParentForm_MdiChildActivate(object sender, EventArgs e)
{
    // Si no hay formularios hijos, se oculta el control con pestañas:
    if (this.ActiveMdiChild == null)
    {
        uxWindowTabs.Visible = false;
    }
    else
    {
        // Si el formulario hijo es nuevo y no tiene pestaña,
        // se crea una nueva pestaña:
        if (this.ActiveMdiChild.Tag == null)
        {
            // Se añade la pestaña con el título del formulario hijo:
            TabPage tp = new TabPage(this.ActiveMdiChild.Text);
            tp.Tag = this.ActiveMdiChild;
            tp.Parent = uxWindowTabs;
            tp.Show();

            this.ActiveMdiChild.Tag = tp;
            this.ActiveMdiChild.FormClosed +=
                new FormClosedEventHandler(ActiveMdiChild_FormClosed);
        }

        uxWindowTabs.SelectedTab = (this.ActiveMdiChild.Tag as TabPage);

        if (!uxWindowTabs.Visible)
        {
            uxWindowTabs.Visible = true;
        }
    }
}
```

Asimismo, a través de la clase Status, que implementa la interfaz INotifyPropertyChanged y cuya única instancia (estática) lanza un evento cada vez que se modifica alguna de las propiedades, las ventanas hijas pueden, modificando las propiedades correspondientes, informar al contenedor MDI de sus cambios de estado, para que éste pueda actualizar la información que transmite al usuario a través de la barra de estado, y activar o desactivar los controles necesarios. La implementación de la clase Status se muestra en la siguiente página.


```

/// <summary>
/// Interfaz para las vistas que escucharán el evento lanzado por esta clase
/// </summary>
internal interface IStatusView
{
    void PropertyChanged(object sender, PropertyChangedEventArgs e);
}
/// <summary>
/// Clase que actúa como intermedia entre el formulario MDI padre y los hijos,
/// para que éstos puedan comunicarse con él sin acceder a él directamente
/// </summary>
internal class Status : INotifyPropertyChanged
{
    internal enum StatusType
    {
        Saving, Searching, Ready
    }

    private static Status _instance = null;

    // Se utiliza el patrón Singleton para no saturar la interfaz gráfica
    internal static Status Instance
    {
        get { return _instance; }
    }

    static Status()
    {
        _instance = new Status();
        _instance.Reset();
    }

    #region Campos
    private bool _sourceLoaded;
    ...
    private StatusType _status;
    #endregion

    #region Propiedades
    internal bool SourceLoaded
    {
        get { return _sourceLoaded; }
        set
        {
            if (_sourceLoaded != value)
            {
                _sourceLoaded = value;
                // Al cambiar la propiedad, se lanza el evento
                OnPropertyChanged(this, new PropertyChangedEventArgs("SourceLoaded"));
            }
        }
    }
    ...

    internal StatusType StatusChanged
    {
        get { return _status; }
        set
        {
            if (_status != value)
            {
                _status = value;
                OnPropertyChanged(this, new PropertyChangedEventArgs("StatusChanged"));
            }
        }
    }
}
#endregion

```

Los tres formularios hijos, DictionaryForm, EditEntryForm y SubtitleForm, heredan de un formulario base MDIChildForm, que les proporciona una serie de propiedades y métodos comunes a los tres, tales como el método de acciones de edición sobre los controles de texto. También lanza un evento, RetrieveEntry, cuando se solicita recuperar el estado o elemento anterior (o siguiente) desde la IGU.

```
public bool RetrieveBack {
    get { return this._retrieveBack; }
}
protected void OnRetrieveEntry(EventArgs e) {
    if (this.RetrieveEntry != null)
        RetrieveEntry(this, e); }
public bool ProcessBack() {
    this._retrieveBack = true;
    OnRetrieveEntry(EventArgs.Empty);
    return true; }

public bool ProcessForward() {
    this._retrieveBack = false;
    OnRetrieveEntry(EventArgs.Empty);
    return true; }

public event EventHandler<EventArgs> RetrieveEntry;
```

Los tres formularios hijos inicializan en su constructor un presentador, al que se pasan a sí mismos como parámetros, así como la instancia de la clase de Servicio. En las siguientes líneas podrá apreciarse su implementación para la clase DictionaryForm.

```
private DictionaryPresenter _presenter; // Presentador
...
private DictionaryForm()
{
    InitializeComponent();
    this._presenter = new DictionaryPresenter(this, DictionaryService.Instance);
}
```

Para proceder a la consulta en el repositorio, los tres formularios lanzan en un nuevo hilo un evento de búsqueda, que será escuchado por el presentador correspondiente. Así se evita bloquear la herramienta mientras se realiza la búsqueda.

```
// Método protegido, encapsula el lanzamiento del evento de búsqueda:
protected void OnFindEntry(SearchEventArgs e)
{
    if (this.FindEntry != null)
    {
        // Se indica el nuevo estado de búsqueda:
        Status.Instance.StatusChanged = Status.StatusType.Searching;
        ThreadStart findStarter = delegate { this.FindEntry(this, e); };
        // Se lanza el evento en un hilo asíncrono, para no paralizar la aplicación:
        Thread newThread = new Thread(findStarter);
        newThread.CurrentUICulture = Config.Instance.DefaultCulture;
        newThread.Start();
    }
}
```

El presentador devuelve el resultado a través de la propiedad EntryDataSource. Puesto que se ha realizado la consulta de forma asíncrona, la plataforma .Net no

permite que modifique los controles creados desde el hilo principal, por lo que debe recurrirse a un `MethodInvoker` que solicite al hilo principal la ejecución del método requerido. A continuación se muestra la implementación para el formulario `EditEntryForm`:

```
// Recibe los resultados de la consulta por parte del presentador
public BOExtendedEntryCollection EntryDataSource {
    set { ShowResults(value); }
}

private void ShowResults(BOExtendedEntryCollection results)
{
    if (uxEntries.InvokeRequired)
    {
        // Se reinvoa el mismo método si es necesario, solicitándole
        // al hilo que creó el control uxEntries que actualice el control
        BeginInvoke(new MethodInvoker(delegate() { ShowResults(results); }));
    }
    else
    {
        this.bOExtendedEntryBindingSource.Clear();
        BOExtendedEntryCollection entryList = results;
        if (results == null)
        {
            MessageBox.Show(Properties.Messages.ErrorAccessingDictionary);
        }
        // Se enlaza la lista de entradas con la colección recibida:
        bOExtendedEntryCollectionBindingSource.DataSource = entryList;

        Status.Instance.StatusChanged = Status.StatusType.Ready;

        // Si la lista está vacía, se borra el contenido mostrado, se
        // desactiva el panel y se muestra la etiqueta correspondiente:
        if (entryList.Count == 0)
        {
            UpdateDisplay(false, true);
            uxSearchBox.Select();
            return;
        }
        // Si no, se carga la primera entrada:
        uxEntries.SetSelected(0, true);
        UpdateDisplay(true, true);

        AddFieldList();
    }
}
```

La ventana de subtitulación se encarga principalmente de solicitar los subtítulos y las búsquedas en el diccionario al presentador, siempre mediante eventos, y mostrarlos correctamente en pantalla. Puede haber varios motivos para solicitar subtítulos al presentador: que el usuario desee ir a un número determinado de subtítulo, que presione las teclas de avance o retroceso de página o de subtítulo, que utilice las barras de desplazamiento, que pulse alguno de los controles del vídeo, o que el avance del vídeo requiera pasar al subtítulo siguiente. También recibe las órdenes de “abrir nuevo archivo”, “cerrar archivo”, etcétera por parte del usuario, y se

las traslada al presentador en los casos que requieran una lógica más compleja. La mayoría de estas interacciones, no obstante, no revisten demasiada complicación.

Es quizá la actualización de los subtítulos en pantalla durante la reproducción del vídeo, que conviene que se realice desde un hilo asíncrono para no bloquear la herramienta, el componente que presenta una complejidad más elevada.

```
// Volatile indica al compilador que varios hilos accederán a este campo:
private volatile bool _stopSubtitleUpdating = false;
private Video _video;

private Thread _updaterThread;
    private int _activeSubtitleIndex = 0; // Índice del subtítulo activo

...

private void StartSubtitleUpdater() {
    if (_video != null && _video.Playing) {
        // Se crea el objeto con el nuevo hilo:
        _updaterThread = new Thread(DisplaySubtitles);
        // Comienza el hilo que refresca los subtítulos en pantalla:
        _updaterThread.Start();
        // Entramos en un bucle hasta que se activa el hilo:
        while (!_updaterThread.IsAlive) ;
    }
}

private void StopSubtitleUpdating() {
    // Si se está viendo el vídeo y actualizando los subtítulos,
    // se detiene el proceso:
    if (_updaterThread != null && _updaterThread.IsAlive) {
        // Se solicita que el hilo de actualización se detenga:
        _stopSubtitleUpdating = true;
        // Se usa el método Join para bloquear el hilo actual hasta que
        // ermene el hilo de actualización.
        _updaterThread.Join();

        if(_activeSubtitles == null)
            ClearDisplayedSubtitles();
    }
}

...

private void UpdateSubtitleText(Control line, string text)
{
    // InvokeRequired compara el identificador del hilo que llama
    // con el del creador del control. Si son distintos, devuelve "true".
    if (line.InvokeRequired)
    {
        SetSubTextCallback d = new SetSubTextCallback(UpdateSubtitleText);
        this.Invoke(d, new object[] { line, text });
    }
    else
    {
        line.SuspendLayout();
        line.Text = text;
        line.Left = (line.Parent.Width / 2) - (line.Width / 2);
        line.ResumeLayout();
    }
}
```

UpdateSubtitleText solicita al hilo creador de los controles de texto que los actualice con el texto indicado. Como se observa en el código fuente, se ha utilizado una variable volátil, _stopSubtitleUpdating, para indicar al hilo de refresco cuándo se le solicita que se detenga. El código de DisplaySubtitles() se muestra a continuación:

```
private void DisplaySubtitles()
{
    TimeSpan currentPosition;
    TimeSpan oldPosition = TimeSpan.Zero;

    // Mientras no se haya cancelado el refresco de subtítulos, se continúa en el bucle:
    while (!_stopSubtitleUpdating)
    {
        currentPosition = TimeSpan.FromSeconds(_video.CurrentPosition);

        // Se actualiza el indicador de tiempo cada 200 ms aproximadamente
        if (currentPosition - oldPosition >= new TimeSpan(0, 0, 0, 0, 200))
        {
            uxCurrentTime.Text = TimeToString(currentPosition);
            oldPosition = currentPosition;
        }

        // Si el vídeo coincide con el subtítulo actual
        if (_activeSubtitles.CurrentIndex(currentPosition) == _activeSubtitleIndex)
        {
            // Si se sitúa entre los códigos de tiempo de entrada y salida,
            // se muestra el texto:
            if (currentPosition.CompareTo(_activeSubtitles[_activeSubtitleIndex].StartTime)
                == 1
                && currentPosition.CompareTo(_activeSubtitles[_activeSubtitleIndex].EndTime)
                == -1)
            {
                if (string.IsNullOrEmpty(uxSubtitle1.Text)
                    || string.IsNullOrEmpty(uxSubtitle2.Text))
                {
                    UpdateSubtitleText(uxSubtitle1,
                        _activeSubtitles[_activeSubtitleIndex].Line1);
                    UpdateSubtitleText(uxSubtitle2,
                        _activeSubtitles[_activeSubtitleIndex].Line2);
                }
            }

            // Si se sitúa tras el código de salida, no se muestra ningún texto:
            if (currentPosition.CompareTo(_activeSubtitles[_activeSubtitleIndex].EndTime)
                == 1)
            {
                if ((uxSubtitle1.Text != string.Empty)
                    || (uxSubtitle2.Text != string.Empty))
                {
                    ClearDisplayedSubtitles();
                }
            }
        }

        // Si no, puede que se haya llegado al siguiente subtítulo:
        else if (_activeSubtitles.CurrentIndex(currentPosition)
            == _activeSubtitleIndex + 1)
        {
            _changeStyle = ChangeStyle.Flow;
            OnMovingToSubtitle(new MoveToEventArgs(_activeSubtitleIndex + 1));
        }

        // Si no, el usuario ha cambiado el subtítulo actual
        // y en este bucle no se hace nada

        Thread.Sleep(33); // Duración en milisegundos de un fotograma
                        // (asumiendo 30 fotogramas por segundo)
    }
    _stopSubtitleUpdating = false;
}
```

7.2.2 Presentación

La capa de presentación se compone de las tres clases de presentación (DictionaryPresenter, EditEntryPresenter y SubtitlePresenter), herederas de PresenterBase, y de una clase Memento genérica (MementoStore<T>) que almacena el historial de entradas consultadas, para EditEntryPresenter y SubtitlePresenter, y los índices de los últimos subtítulos visitados, para SubtitlePresenter.

También contiene las interfaces que han de implementar las respectivas vistas, y las clases de argumentos herederas de EventArgs que se pasan en distintos eventos lanzados desde las vistas.

Todas las vistas implementan una interfaz que hereda de IDictionaryBaseView<T>, por lo que comparten los siguientes elementos:

```
public interface IDictionaryBaseView<T>
{
    bool RetrieveBack { get; }
    T EntryDataSource { set; }
    bool SetBack { set; }
    bool SetForward { set; }
    bool IsSaved { set; }

    event EventHandler<SearchEventArgs> FindEntry;
    event EventHandler<EventArgs> RetrieveEntry;
}
```

El evento FindEntry solicita la búsqueda en el diccionario, y RetrieveEntry solicita la recuperación de una entrada del historial. El gestor del evento FindEntry se mantiene como un método abstracto en esta clase, encargándose las herederas de su implementación.

EntryDataSource devuelve a la GUI los objetos solicitados por esta. Dentro de la clase PresenterBase<T>, se pueden destacar la lista _fastList<T> de los últimos elementos visitados, y el gestor de evento View_RetrieveEntry():

```
// Lista con los últimos elementos visitados:
protected MementoStore<T> _fastList = new MementoStore<T>();
...
/// <summary>
/// Recupera el elemento anterior o posterior de la lista de elementos visitados,
/// en función del valor de this.View.RetrieveBack
/// </summary>
/// <param name="sender">Objeto que ha lanzado el evento</param>
/// <param name="e">Argumentos</param>
protected void View_Retrieve(object sender, EventArgs e)
{
    int currentIndex;

    if (this.View.RetrieveBack) // Se debe recuperar el elemento anterior
    {
        this.View.EntryDataSource = _fastList.RetrieveElement(true);
        currentIndex = _fastList.CurrentIndex;
        if (currentIndex == 0)
            this.View.SetBack = false;
        if (currentIndex == _fastList.Count - 2)
            this.View.SetForward = true;
    }
    else // Se debe recuperar el elemento posterior
    {
        this.View.EntryDataSource = _fastList.RetrieveElement(false);
        currentIndex = _fastList.CurrentIndex;

        // Si se ha llegado al último elemento de la lista,
        // se notifica a la interfaz gráfica:
        if (currentIndex == _fastList.Count - 1)
            this.View.SetForward = false;
        if (currentIndex == 1)
            this.View.SetBack = true;
    }
}
```

Todos los presentadores mantienen una estructura similar, muy centrada en los gestores de los eventos lanzados por la GUI. Estos se encargan de realizar búsquedas en el diccionario, persistir los objetos de negocios, abrir y guardar archivos (en el caso del presentador de subtítulos), presentar a la vista los objetos solicitados... El resto de campos y métodos cumplen funciones auxiliares de los anteriores, almacenando datos o realizando parte de la lógica.

Se mostrará aquí, a modo de ejemplo, la implementación que del método `View_FindEntry()` realiza `DictionaryPresenter`:

```
/// <summary>
/// Busca el texto proporcionado por el usuario y devuelve el resultado a través
/// de la propiedad EntryDataSource
/// de la vista.
/// </summary>
/// <param name="sender">Objeto que lanza el evento</param>
/// <param name="e">Argumentos de búsqueda</param>
protected override void View_Find(object sender, SearchEventArgs e)
{
    lock (_syncFindObj)
    {
        // Se configuran los criterios de búsqueda:
        Criteria searchCriteria = Criteria.Create(true,
            true, e.SearchPOS, e.SearchField);

        if (e.SecondaryLanguage)
        {
            searchCriteria.Language1 = Config.Instance.SecondaryLanguage;
            searchCriteria.Language2 = Config.Instance.PrimaryLanguage;
        }
        else
        {
            searchCriteria.Language1 = Config.Instance.PrimaryLanguage;
            searchCriteria.Language2 = Config.Instance.SecondaryLanguage;
        }
        // Se realiza la búsqueda:
        _entryList = this._service.Find(e.SearchString, searchCriteria);

        // Si ha habido un problema accediendo a la base de datos, se informa
        // a la Vista:
        if (_entryList == null)
            this.View.EntryDataSource = null;

        else
        {
            // Se procesa la lista recibida en el almacén de elementos anteriores:
            if (_entryList.Count > 0)
            {
                _fastList.ProcessElement(_entryList);
                if (_fastList.Count > 1)
                    this.View.SetBack = true;
            }
            // Y se devuelve el resultado a la vista:
            this.View.EntryDataSource = _entryList as BOExtendedEntryCollection;
        }
    }
}
```

7.2.3 Negocio

En esta capa se sitúan, principalmente, los objetos de negocios que contendrán, por un lado, las entradas devueltas por el diccionario, y por otro, los subtítulos con los que trabaja el usuario. También hay aquí clases auxiliares como la clase *Criteria* (criterios de búsqueda), que se pasa como argumento desde los presentadores hasta la capa de acceso a datos en cada consulta; la clase *Validation*, de la cual declara una instancia cada objeto de negocios, para validar que las propiedades introducidas son correctas; o la estructura *VideoKey*, empleada por la clase *BOSubtitles* para asignar códigos de tiempo a cada índice de subtítulo.

Al modificar cualquier propiedad de los objetos de negocios se mantiene constancia de ello a través del método `DataStateChanged()`, de modo que el objeto pueda comunicar su estado en todo momento, a través de propiedades públicas como `IsDirty` o `IsNew`. Así, la aplicación, desde los presentadores y la GUI, podrá alertar al usuario si, por ejemplo, va a cerrar un formulario con un objeto modificado.

En la siguiente propiedad de la clase `BOEntry` se puede observar el uso de la instancia de validación y del método mencionado en el párrafo anterior:

```
/// <summary>
/// Una cadena con el idioma de la expresión (dos caracteres) y,
/// opcionalmente, el país en que se emplea (otros dos caracteres).
/// Sigue la práctica recomendada por la ontología Dublin Core, RFC 1766
/// </summary>
public string Language
{
    get
    {
        return _language;
    }
    set
    {
        if (_language != value)
        {
            string propertyName = "Language";
            ValidationInstance.ValidateClear(propertyName);
            // Se requiere que la propiedad tenga un valor
            ValidationInstance.ValidateRequired(propertyName,
                Config.Instance.Language, value);
            // Se requiere una longitud máxima de 5 caracteres
            ValidationInstance.ValidateMaxLength(propertyName,
                Config.Instance.Language, value, 5);
            // Y una longitud mínima de 2 caracteres
            ValidationInstance.ValidateMinLength(propertyName,
                Config.Instance.Language, value, 2);
            this.DataStateChanged(EntityStateEnum.Modified);
            _language = value;
        }
    }
}
```

Todas las clases de negocios heredan de la clase `BOBase`, que implementa la interfaz `IDataErrorInfo`, por lo que si no se cumplen los requerimientos impuestos por los métodos de `ValidationInstance`, los controles de la IGU enlazados a dichos objetos mostrarán un mensaje no invasivo de error al usuario, indicándole cómo debe modificar los datos para que sean válidos.

La clase `BOSubtitles` almacena la colección de subtítulos leídos desde un archivo de texto, mediante el método estático `LoadFromFile()`. Para proceder a su lectura, se recurre a las expresiones regulares. A continuación se muestra parte del código de dicho método, por considerarse de especial interés:

```

/// <summary>
/// Carga una lista de subtítulos desde un archivo
/// </summary>
/// <param name="fileName">Nombre del archivo que contiene los subtítulos</param>
/// <returns>Lista de subtítulos</returns>
/// <remarks>No se controlan las excepciones aquí (posibles si el archivo está dañado
/// o no cumple el formato), ya que no se van a gestionar desde aquí.
/// <remarks>Será la IGU la que las gestione, informando al usuario</remarks>
public static BOSubtitles LoadFromFile(string fileName)
{
    if (!File.Exists(fileName))
    {
        throw (new FileNotFoundException());
    }

    BOSubtitles subtitleList = new BOSubtitles();

    using (StreamReader sr = new StreamReader(fileName))
    {
        Regex reNumber = new Regex(@"\d{1,4}"); // Expresión regular de número
        string time = @"^(?<startTime>\d{2}:\d{2}:\d{2}\.\d{3})\"
            + @"-(?<endTime>\d{2}:\d{2}:\d{2}\.\d{3})$";
        Regex reTime = new Regex(time); // Expresión regular de códigos de tiempo

        // Más expresiones regulares
        ...

        TimeSpan previousStart = TimeSpan.Zero;
        while (sr.Peek() != -1)
        {
            //Se lee el número de subtítulo:
            Match ma = reNumber.Match(sr.ReadLine());
            int subIDRead = Convert.ToInt32(ma.Value);
            // Se leen los tiempos de entrada y salida del subtítulo...
            ma = reTime.Match(sr.ReadLine());
            TimeSpan.TryParse(ma.Groups["endTime"].Value, out endTimeRead);
            BOSubtitle subtitle = BOSubtitle.Create(subIDRead,
                startTimeRead, endTimeRead);
            // Se lee el número de líneas, marcas, negrita y cursiva...
            string line = sr.ReadLine();
            ...
            // Se almacenan los comentarios, de haberlos,
            // y la/s línea/s que componen el subtítulo
            ...
            // Se marca el subtítulo como no modificado
            subtitle.ResetState();
            subtitle.Modified
                += new EventHandler<EventArgs>(subtitleList.SubtitleModified);
            subtitleList.Add(subtitle);
            if (subtitleList.Count > 1)
            {
                // Se emparejan los códigos de tiempo del subtítulo con su índice
                // en la colección, y se añaden al diccionario _keyStore
                subtitleList.AddKey(new VideoKey(previousStart,
                    subtitle.StartTime, subtitleList.IndexOf(subtitle) - 1);
                previousStart = subtitle.StartTime;
            }
            if (sr.Peek() == -1)
            {
                // En el último subtítulo, se pone un código de salida que garantice
                // su duración hasta el final del vídeo (100 horas)
                subtitleList.AddKey(new VideoKey(subtitle.StartTime,
                    TimeSpan.FromHours(100)), subtitleList.IndexOf(subtitle));
            }
            subtitleList.SubtitleFile = fileName;
        }
        return subtitleList;
    }
}

```

7.2.4 Servicio

Desde esta capa se llama, por medio de la instancia de la clase DictionaryService, a los métodos de consulta y persistencia en el repositorio de la clase DALEntry, y se devuelve el resultado a la capa de presentación.

```
#region Campos, propiedades y constructor
private static DictionaryService _instance = null;

// Patrón Singleton
public static DictionaryService Instance
{
    get { return _instance; }
}

static DictionaryService()
{
    _instance = new DictionaryService();
}
#endregion

#region Métodos públicos

// Se persisten las modificaciones realizadas en la entrada activeEntry
public bool Save(BOExtendedEntry oldActiveEntry,
    BOExtendedEntry activeEntry)
{
    // DALEntry dataAccess = new DALEntry();

    // Si la entrada es nueva, se invoca al método Agregar
    // de la base de datos
    if (activeEntry.IsNew)
    {
        return DALEntry.Instance.Add(activeEntry);
    }
    // Si no, se llama al método Modificar
    else
        return DALEntry.Instance.Modify(oldActiveEntry, activeEntry);
}

// Borra la entrada entry
public bool Delete(BOExtendedEntry entry)
{
    return DALEntry.Instance.Delete(entry);
}

// Busca la cadena searchString en el repositorio,
// según los criterios de consulta especificados
public BOExtendedEntryCollection Find(string searchString,
    Criteria searchCriteria)
{
    BOExtendedEntryCollection result
        = DALEntry.Instance.Find(searchString, searchCriteria);

    return result;
}
#endregion
```

7.2.5 Acceso a datos

Desde aquí se accede a la base de datos para persistir las entradas en el repositorio. Para ello se hace uso de la biblioteca de clases SemWeb.Net, que proporciona utilidades para construir tripletas RDF y almacenarlas en archivos o bases de datos, así como proceder a la búsqueda de información RDF por medio de consultas SPARQL.

```
/// <summary>
/// Busca la cadena de búsqueda especificada por el usuario
/// </summary>
/// <param name="searchString">Cadena de búsqueda</param>
/// <param name="searchCriteria">Criterios que se aplicarán en la búsqueda</param>
/// <returns>Colección de entradas encontradas en el diccionario</returns>
public BOExtendedEntryCollection Find(string searchString, Criteria searchCriteria)
{
    // Se eliminan los caracteres inadecuados:
    StringBuilder sb = new StringBuilder();
    for (int i = 0; i < searchString.Length; i++)
        if ((char.IsLetterOrDigit(searchString[i])) || searchString[i] == '-')
            sb.Append(searchString[i]);
        else sb.Append(" ");

    // Se crea la cadena de consulta:
    string query = BuildQuery(sb.ToString(), searchCriteria);

    // Se carga el diccionario:
    Store dictionary;
    try {
        if (Config.Instance.UseFile)
            dictionary = GetDictionary(1);
        else dictionary = GetDictionary(DICTIONARY_ID);
    } catch (Exception)
    {
        return null;
    }

    // Se crea el sumidero de resultados:
    EntryQuerySink sink = new EntryQuerySink();
    // Se crea la consulta con la cadena de consulta...
    Query sparqlQuery;
    try {
        sparqlQuery = new SparqlEngine(query);
    } catch (QueryFormatException ex)
    {
        throw new ArgumentException("", ex);
    }

    // ...y se ejecuta:
    try {
        sparqlQuery.Run(dictionary, sink);
    } catch (QueryExecutionException)
    {
        return null;
    } finally
    {
        dictionary.Dispose();
    }

    // Se devuelve una colección de entradas BOExtendedEntryCollection
    // con el resultado devuelto por el diccionario:
    return PopulateEntryList(sink.EntryTable, searchCriteria.MaxResults);
}
```

En el código anterior puede observarse todo el proceso de búsqueda. Primero se “limpia la cadena”, luego se crea la cadena de consulta SPARQL llamando al método

privado BuildQuery() y se carga el diccionario. A continuación se crea un sumidero de resultados de la clase EntryQuerySink y se crea el objeto de consulta SPARQL a partir de la cadena ya creada. Finalmente, se ejecuta la consulta en el diccionario, que introduce los resultados en el sumidero, se puebla la lista de entradas BOExtendedEntry con los resultados obtenidos, y se devuelven al llamante.

La clase EntryQuerySink hereda de la clase QueryResultSink, incluida en SemWeb.Net. Sobrescribe el método Add() (cuyo código se muestra abajo) para recibir cada resultado de la consulta, que añade a la tabla de datos _entryTable. De vuelta en DALEntry, se convierte esa tabla de datos sin formato en una lista BOExtendedEntryCollection mediante el método PopulateEntryList(), dotándolos de significado para la aplicación.

```
/// <summary>
/// Añade cada resultado de la consulta a una fila de la Tabla de Datos,
/// con cada variable en la columna correspondiente
/// </summary>
/// <param name="result">Resultado de la consulta SPARQL</param>
/// <returns>true</returns>
public override bool Add(VariableBindings result)
{
    // Se crea una nueva fila con el mismo esquema de la tabla _entryTable
    DataRow newRow = _entryTable.NewRow();

    // Se añade cada variable del resultado en la columna
    // correspondiente de la tabla _entryTable:
    foreach (Variable var in result.Variables)
    {
        if (var.LocalName != null && result[var] != null)
        {
            if ((var.LocalName == DALConstants.RESULT_POS)
                || (var.LocalName == DALConstants.RESULT_EQEXPRESSIONTYPE)
                || (var.LocalName == DALConstants.RESULT_EXPRESSIONTYPE))
            {
                string type;
                // Obtener el tipo de expresión o de categoría gramatical
                ...
                newRow[var.LocalName] = type;
            }
            else
            {
                char[] trimChars = { '\'"' };
                newRow[var.LocalName] = result[var].ToString().Trim(trimChars);
            }
        }
    }
    _entryTable.Rows.Add(newRow);
    return true;
}
```

A la inversa, para almacenar datos en el repositorio, se toma el objeto BOExtendedEntry y, opcionalmente, el mismo objeto antes de su modificación, y se construyen las tripletas que se van a añadir o eliminar del diccionario. Para ello se

crean instancias de la clase `Statement`, definida en `SemWeb.Net`. A continuación se mostrará un breve fragmento de código a modo de ejemplo, del método `AddEntry()`.

```
/// <summary>
/// Añade las oraciones correspondientes a la entrada
/// a la lista de oraciones especificada
/// </summary>
/// <param name="entry">Entrada a añadir</param>
/// <param name="statementList">Lista de oraciones a la que se añadirán
/// las nuevas</param>
/// <param name="entity">URI con el sujeto de las oraciones</param>
private void AddEntry(BOEntry entry,
    List<Statement> statementList, Entity entityURI)
{
    string prefix;

    // Se añaden a la lista de oraciones los atributos de la entrada:
    statementList.Add(new Statement
        (entityURI, DALConstants.LABEL, (Literal)entry.Label));
    statementList.Add(new Statement
        (entityURI, DALConstants.LANGUAGE, (Literal)entry.Language));

    if ((entry.EntryType == ExpressionType.Word.ToString())
        || (entry.EntryType == ExpressionType.Phrase.ToString()))
        prefix = DALConstants.SUMO;
    else
        prefix = DALConstants.EX;
    statementList.Add(new Statement
        (entityURI, DALConstants.TYPE, (Entity)(prefix + entry.EntryType)));

    // Se comprueba si no está vacía la propiedad pasada como argumento,
    // en cuyo caso se añade a la lista de oraciones:
    AddCheckedStatement(statementList,
        entityURI, DALConstants.NOTES, entry.Notes);
    AddCheckedStatement(statementList,
        entityURI, DALConstants.ALTSPELLING, entry.AltSpelling);
}
```

8. Evaluación de la herramienta

Puesto que la clienta es la misma autora de la herramienta de traducción, se ha partido de una cierta ventaja a la hora de ajustarse a sus exigencias. Sin embargo, esto no garantiza la usabilidad de la aplicación, por lo que conviene hacer una evaluación exhaustiva para comprobar a posteriori qué elementos son adecuados y qué aspectos son susceptibles de mejoras.

Para conocer los detalles de uso de la aplicación, véase el Apéndice B: Manual de usuario.

Tras dedicar un tiempo razonable a evaluar todas las funciones de la herramienta, la clienta realiza las siguientes observaciones:

- Se echa en falta una búsqueda más inteligente en el diccionario, que tenga en cuenta formas verbales o terminaciones de género y número, ya que al realizar búsquedas de términos en la pantalla de subtitulación, lo habitual es que las palabras no aparezcan en su forma básica.
- Sería deseable poder ajustar la dificultad del texto o el nivel de lectura del público objetivo en el formulario de preferencias, para no ceñirse siempre exclusivamente a la regla de los seis segundos. En ocasiones convendría limitar aún más el espacio, mientras que en otras será posible incluir más información en el subtítulo, aunque quede más largo.
- Aunque las reducidas dimensiones del monitor de la clienta no permiten aprovechar al máximo la funcionalidad MDI, se considera un aspecto útil para una pantalla de mayor resolución, permitiendo modificar o consultar el diccionario sin cambiar de ventana. No obstante, se observa que la barra de pestañas permite en todo caso saltar cómodamente de una ventana a otra.
- Se considera muy práctico poder modificar las opciones más habituales, como el modo de trabajo o la búsqueda automática, pulsando simplemente un botón en la barra de estado.
- Se valora muy positivamente la extracción por lotes de entradas en la pantalla de subtitulación. Permite introducir rápidamente en el diccionario expresiones de la propia traducción una vez realizada, o de traducciones realizadas por otras personas, si se tiene acceso a los archivos. De esta forma, cuando se trata de la propia traducción, el traductor puede continuar con ésta sin más que marcar el subtítulo y dejar para más adelante la introducción de los términos en el diccionario. Sí que se consideraría más práctico emplear otro tipo de marca para

esta función y diferenciarla de la marca existente, que se emplea también para revisar subtítulos confusos o inacabados.

- Inicialmente no es tan patente la ventaja que ofrece la búsqueda en el diccionario desde la misma ventana de traducción, puesto que el repositorio contiene una cantidad muy reducida de entradas. Sin embargo, resulta evidente que a medida que se introduzcan más términos y expresiones, el tiempo empleado para hacerlo, en parte gracias a la herramienta de extracción por lotes ya comentada, se verá claramente compensado por el ahorro que supondrá poder acceder al repositorio directamente, sobre todo si se utiliza la opción de búsqueda automática en el diccionario al entrar en cada subtítulo.
- A la clienta le resulta especialmente interesante que la consulta automática se realice sobre un diccionario personalizado. Efectuar consultas automáticas de cada subtítulo en un diccionario general saturaría el cuadro de sugerencias de forma innecesaria (dado que al traductor se le presupone un buen nivel de ambos idiomas de trabajo), y de esta forma se ajusta a las necesidades reales de cada traductor.
- Sería deseable poder personalizar los campos que se incluyen en las entradas (por ejemplo: fuente de la que se ha obtenido).
- La búsqueda no segura parece útil para establecer nuevas relaciones entre expresiones inicialmente, aunque la clienta no está convencida de que sea práctica cuando el diccionario alcance un tamaño considerable.

En cualquier caso, la clienta se muestra satisfecha con la herramienta y considera que es, a diferencia de otras existentes, un programa de subtitulación realmente orientado al traductor. La integración del repositorio terminológico en la solución y la extracción por lotes de expresiones le parecen sus dos características más relevantes y novedosas.

9. Trabajos futuros y conclusiones

9.1 Trabajos futuros

Aunque ha tratado de desarrollarse una solución lo más completa posible, es inevitable e, incluso, deseable que esta primera herramienta dé pie a nuevas ampliaciones y mejoras, con objeto de cubrir todas las necesidades que pueda tener el traductor de subtítulos.

A continuación se enumeran varias líneas de trabajo que podrían seguirse, en orden de mayor a menor interés para la autora de este documento:

- **Acceso multiusuario al repositorio:** aunque contar con un repositorio terminológico personal es extremadamente útil para el traductor, también sería muy ventajoso poder compartir el repositorio entre una comunidad de usuarios (traductores de una misma agencia o estudio, miembros de una asociación de traductores audiovisuales...). Esto permitiría el crecimiento del diccionario a mucha más velocidad y supondría mucho menos trabajo para cada traductor individual. Para mantener un nivel de calidad adecuado, habría que garantizar que solo los traductores profesionales tuvieran acceso a él.
- **Importar y exportar los formatos de archivos de subtítulos estándares en la industria:** puesto que el proyecto se ha centrado en el proceso de traducción y consulta del diccionario, los formatos de subtítulos con los que trabaja actualmente son muy reducidos, limitándose al formato propio creado con la aplicación (.ats), y el formato .srt. Se ha incluido una interfaz, IParsable, que permitirá que futuras clases encargadas de importar y exportar subtítulos en otros formatos se integren sin problemas en la aplicación al implementarla.
- **Modificar los códigos de tiempo:** en este proyecto se ha partido del requisito específico de no permitir modificar los códigos de tiempo del archivo de subtítulos, ya que no se considera deseable que el traductor modifique el archivo maestro de origen, por lo que pautar solo el de destino supondría la no alineación de los subtítulos en las columnas fuente y destino. No obstante, empleando técnicas adecuadas de posicionamiento de cada subtítulo y de búsqueda automática en el diccionario en función de los códigos de tiempo, podría proporcionarse esta utilidad. De esta forma, el traductor tendría la libertad de no traducir partes innecesarias de los subtítulos (por ejemplo, si se dice algo en su propio idioma) o combinar o separar subtítulos cuando resultase así más natural su lectura. Otra opción más sencilla sería permitir la modificación de los códigos de tiempo en la

última etapa, una vez realizada la traducción, desactivando la búsqueda automática e incluso cerrando el archivo fuente.

- **Otras mejoras:** otras posibles adiciones, quizá de menor calado que las anteriores, pero también útiles, podrían ser:
 - **Copiar y pegar un rango de subtítulos:** ya que a veces se repite el texto de subtítulos anteriores, por ejemplo en estribillos de canciones, recuerdos, viajes en el tiempo...
 - **Más opciones de formato en el subtítulo:** como tipo de letra, color o tamaño, posición en pantalla, etcétera. También más tipos de marca, tal como se comentó en el apartado de evaluación de la herramienta.
 - **Reglas lingüísticas por idioma:** sería interesante agregar juegos de reglas lingüísticas por idioma que filtren el texto antes de hacer la búsqueda y permitan buscar infinitivos a partir de una forma verbal, singulares a partir del plural, etcétera.

9.2 Conclusiones

Una vez finalizado el proyecto, conviene analizar si se ha logrado un correcto cumplimiento de los objetivos marcados en la introducción. Repasando dichos objetivos, se puede confirmar que el proyecto realizado satisface adecuadamente todos los requisitos planteados. Así, se ha desarrollado una herramienta de traducción asistida de subtítulos que proporciona al usuario un repositorio terminológico integrado en la solución. Más específicamente:

1. Le proporciona un diccionario de términos y expresiones entre la lengua de origen y la lengua meta, que podrá crear y modificar para ajustarlo a sus necesidades individuales.
2. En este mismo diccionario puede incluir expresiones en la lengua meta que le serán de utilidad durante la revisión y mejora de la traducción.
3. Un elemento clave de la herramienta es que las consultas se pueden realizar de forma automática, simplemente entrando en cada subtítulo (y mediante un hilo independiente), de forma que el traductor puede mantener el ritmo de trabajo sin interrupciones.
4. La información que se incluye sobre cada relación semántica en el repositorio incluye contextos, ejemplos, ámbitos y categoría gramatical, siendo las dos primeras opcionales y únicamente para completar la información en caso de que el traductor lo considere necesario.

5. La implementación se ha hecho de manera que en un futuro pueda accederse a un diccionario online compartido sin realizar modificaciones importantes al código.
6. Se ha empleado una ontología RDF para modelar el dominio lingüístico, utilizando predicados y clases ya existentes en los casos en los que éstos estaban disponibles, para facilitar el intercambio de la información incluida en el repositorio y su reutilización en otras aplicaciones, no necesariamente relacionadas con el mundo audiovisual.
7. La modularidad y la arquitectura multicapa de la solución simplificarán las ampliaciones y modificaciones que puedan realizarse en el futuro.

Pese a todo, las propuestas hechas en la evaluación de la herramienta y en el apartado de trabajos futuros, así como la constatación de la poca uniformidad que todavía existe en la práctica profesional, muestran que queda aún funcionalidad por añadir para que la solución pueda ser empleada en contextos lo más diversos posibles.

En cualquier caso, se trata de una solución útil y completa ya en su presente estado, pensada para mejorar la eficiencia y simplificar el trabajo del traductor, proporcionándole las utilidades más necesarias y eliminando en lo posible los elementos superfluos que pudieran lastrar el proceso de traducción. Se ha tenido muy en cuenta la forma de trabajo de este profesional y se ha tratado de adecuar la herramienta a sus necesidades, dotándola además de un manejo sencillo e intuitivo.

La integración del repositorio terminológico en el entorno de trabajo, muy especialmente, sitúa al fin la traducción audiovisual entre las demás modalidades traductorales que se han visto beneficiadas por los avances tecnológicos desde la perspectiva de la asistencia a la traducción. Se cubre así el vacío existente en la actualidad en las herramientas de subtitulación, más enfocadas en general a la creación y el pautado de los subtítulos. La funcionalidad orientada específicamente al aspecto de traducción supone una verdadera novedad para los profesionales del campo, y desde aquí se confía en que sea acogida por estos con el mismo entusiasmo con el que se ha entregado la autora a su desarrollo.

10. Bibliografía

1. **SDL Trados.** [En línea] <http://www.trados.com/>.
2. **Wordfast LLC.** [En línea] <http://www.wordfast.net/>.
3. **OmegaT.** [En línea] <http://www.omegat.org/es/omegat.html>.
4. **World Wide Web Consortium.** RDF Primer. [En línea] <http://www.w3.org/TR/2004/REC-rdf-primer-20040210/>.
5. **Craciunescu, Olivia, Gerding-Salas, Constanza and Stringer-O'Keeffe, Susan.** Machine Translation and Computer-Assisted Translation: a New Way of Translating? *Translation Journal*. 2004, Vol. 8, 3.
6. **Bowker, Lynne.** *Computer-aided translation technology: a practical introduction*. s.l. : University of Ottawa Press, 2002. ISBN 0-7766-3016-4.
7. **Wilkinson, Michael.** Using a Specialized Corpus to Improve Translation Quality. *Translation Journal*. 2005, Vol. 9, 3.
8. *Technology and Translation (a Pedagogical Overview)*. **Biau-Gil, José Ramón y Pym, Anthony.** Tarragona : s.n., 2006.
9. **Centro de Traducción de los Órganos de la Unión Europea.** InterActive Terminology for Europe. [En línea] <http://iate.europa.eu/>.
10. **Díaz Cintas, Jorge.** *Teoría y práctica de la subtitulación Inglés - Español*. s.l. : Ariel Cine, 2003.
11. **TV2003.** [En línea] <http://www.titlevision.dk/prices.htm>.
12. **MacCaption.** <http://www.cpcweb.com/subtitle/>. [En línea]
13. **EZTitles.** <http://www.eztitles.com/>. [En línea]
14. **PoliScript 3000.** <http://www.screen.subtitling.com/index.asp?PageID=18>. [En línea]
15. **WinCaps.** [En línea] http://www.sysmedia.com/subtitling/offline_subtitling.asp.



16. **Swift Create.** [En línea] <http://www.softel.co.uk/>.
17. **Subeasy.** <http://www.mss.gr/subeasy.htm>. [En línea]
18. **Subbits.** [En línea] <http://www.videotoolshed.com/?page=products&pID=12>.
19. **Subtitle Translation Wizard.** [En línea] <http://www.upredsun.com/subtitle-translation/subtitle-translation.html>.
20. **Subtitle Workshop.** [En línea] <http://www.urusoft.net/>.
21. **dotSUB.** [En línea] <http://dotsub.com/>.
22. **Massachusetts Institute of Technology.** Knowledge Representation. [En línea] 1999. <http://www.cs.umbc.edu/courses/771/papers/hayesp.html>.
23. **Barski, Conrad.** How To Tell Stuff To a Computer: The Enigmatic Art of Knowledge Representation. [En línea] <http://www.lisperati.com/tellstuff/index.html>.
24. **World Wide Web Consortium.** Extensible Markup Language (XML). [En línea] <http://www.w3.org/XML/>.
25. **Raggett, Dave.** Getting started with HTML. [En línea] <http://www.w3.org/MarkUp/Guide/>.
26. **Berners-Lee, Tim.** Why RDF model is different from the XML model. [En línea] 1998. <http://www.w3.org/DesignIssues/RDF-XML.html>.
27. **Tauberer, Joshua.** What is RDF and what is it good for? [En línea] <http://www.rdfabout.com/intro/?section=contents>.
28. **Ebiquity Group.** Swoogle Semantic Web Search Engine. [En línea] <http://swoogle.umbc.edu/>.
29. **Obitko, Marek.** RDF Graph and Syntax. [En línea] 2007. <http://www.obitko.com/tutorials/ontologies-semantic-web/rdf-graph-and-syntax.html>.
30. **World Wide Web Consortium.** Primer: Getting into RDF & Semantic Web using N3. [En línea] <http://www.w3.org/2000/10/swap/Primer>.
31. **Gruber, Thomas R.** Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies*. 1995, Vol. 43, 5-6.

32. **Borst, William.** *Construction of Engineering Ontologies for Knowledge Sharing and Reuse.* 1997.
33. **Noy, Natalya F. y McGuinness, Deborah L.** *Ontology Development 101: A Guide to Creating Your First Ontology.* Stanford University, Stanford, California : s.n., 2001.
34. **Gómez-Pérez, Asunción.** *Ontological Engineering: A State of the Art.* Universidad Politécnica de Madrid : s.n.
35. **World Wide Web Consortium.** SPARQL Query Language for RDF. [En línea] 2008. <http://www.w3.org/TR/rdf-sparql-query/>.
36. Apuntes de la asignatura "Metodología de Desarrollo Visual". *Ingeniería Informática, Universidad Carlos III de Madrid.* 2006.
37. **Tauberer, Joshua.** SemWeb.NET: Semantic Web/RDF Library for C#.NET. [En línea] <http://razor.occams.info/code/semweb/>.
38. **InfoJobs.net.** InfoJobs Trends Salarios. [En línea] <http://salarios.infojobs.net/>.
39. **Fowler, Martin.** Supervising Controller. [En línea] <http://www.martinfowler.com/eaaDev/SupervisingPresenter.html>.
40. Dublin Core Metadata Initiative. [En línea] <http://dublincore.org/>.
41. **World Wide Web Consortium.** Simple Knowledge Organization System Reference. [En línea] <http://www.w3.org/TR/2009/REC-skos-reference-20090818/>.
42. **IEEE.** The Suggested Upper Merged Ontology (SUMO) - Ontology Portal. [En línea] <http://www.ontologyportal.org/>.
43. **World Wide Web Consortium.** Defining N-ary Relations on the Semantic Web. [En línea] <http://www.w3.org/TR/swbp-n-aryRelations/>.
44. **Microsoft.** Microsoft Visual Studio Express. [En línea] <http://www.microsoft.com/express/windows/>.
45. —. Información general y conceptual sobre .Net Framework. [En línea] <http://msdn.microsoft.com/es-es/library/zw4w595w.aspx>.



46. **Stanford Center for Biomedical Informatics Research.** Protégé Ontology Editor and Knowledge Acquisition System. [En línea] <http://protege.stanford.edu/>.

47. Getting started with HTML. [En línea] 2005. <http://www.w3.org/MarkUp/Guide/>.

11. Apéndices

11.1 Apéndice A: Consultas SPARQL

CONSULTA 1: CONSULTA AUTOMÁTICA DESDE LA VENTANA DE SUBTITULACIÓN

Ejemplo de consulta inglés->castellano, únicamente de traducciones, donde se busca cualquier palabra.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX ex: <http://example.org/>
PREFIX mydict: <http://mydictionary.org/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX sumo:
<http://www.ontologyportal.org/translations/SUMO.owl.txt#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT ?origLabel ?origLanguage ?origNotes
?origAltSpelling ?origEType
?eqLabel ?eqLanguage ?eqNotes ?eqAltSpelling ?eqEType
?partOfSpeech ?field ?context ?example
WHERE      { ?orig rdfs:label ?origLabel ;
dc:language ?origLanguage ;
rdf:type ?origEType ;
ex:hasEquivalent ?eq .

?eq rdf:Value ?eqValue .
?eq ex:hasEquivalencePOS ?partOfSpeech .
?eq ex:hasEquivalenceField ?field .
?eqValue rdfs:label ?eqLabel ;
      dc:language ?eqLanguage ;
      rdf:type ?eqEType .
?orig dc:language "ES" .
{ ?eqValue dc:language "ES" . }
OPTIONAL { ?orig skos:note ?origNotes . }
OPTIONAL { ?orig skos:altSpelling ?origAltSpelling . }
OPTIONAL { ?eqValue skos:altSpelling ?eqAltSpelling . }
OPTIONAL { ?eqValue skos:note ?eqNotes . }
OPTIONAL { ?eq ex:hasEquivalenceContext ?context . }
OPTIONAL { ?eq ex:hasEquivalenceExample ?example . }
FILTER regex(?origLabel, ".*far.*|.*good.*", "i" )
}
ORDER BY ?origLabel ?eqLanguage ?partOfSpeech ?eqLabel
```


CONSULTA 2: CONSULTA DESDE LA VENTANA DE EDICIÓN DEL DICCIONARIO

Ejemplo de consulta en ambas lenguas de trabajo, tanto sinónimos como traducciones, se busca expresión exacta.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX ex: <http://example.org/>
PREFIX mydict: <http://mydictionary.org/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX sumo: <http://www.ontologyportal.org/translations/SUMO.owl.txt#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT ?origLabel ?origLanguage ?origNotes ?origAltSpelling ?origEType
?eqLabel ?eqLanguage ?eqNotes ?eqAltSpelling ?eqEType ?partOfSpeech ?field ?context
?example
WHERE { ?orig rdfs:label ?origLabel ;
dc:language ?origLanguage ;
rdf:type ?origEType ;
ex:hasEquivalent ?eq .

?eq rdf:Value ?eqValue .
?eq ex:hasEquivalencePOS ?partOfSpeech .
?eq ex:hasEquivalenceField ?field .

?eqValue rdfs:label ?eqLabel ;
dc:language ?eqLanguage ;
rdf:type ?eqEType .
{ ?orig dc:language "ES" . } UNION { ?orig dc:language "EN" . }
{ ?eqValue dc:language "ES" . }
UNION { ?eqValue dc:language "EN" . }
OPTIONAL { ?orig skos:note ?origNotes . }
OPTIONAL { ?orig skos:altSpelling ?origAltSpelling . }
OPTIONAL { ?eqValue skos:altSpelling ?eqAltSpelling . }
OPTIONAL { ?eqValue skos:note ?eqNotes . }
OPTIONAL { ?eq ex:hasEquivalenceContext ?context . }
OPTIONAL { ?eq ex:hasEquivalenceExample ?example . }
FILTER regex(?origLabel, "so far so good", "i" )
}
ORDER BY ?origLabel ?eqLanguage ?partOfSpeech ?eqLabel
```

CONSULTA 3: CONSULTA DESDE LA VENTANA DE CONSULTA DEL DICCIONARIO

Ejemplo de consulta en una lengua de trabajo, tanto de sinónimos como de traducciones, donde se buscan todas las palabras.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX ex: <http://example.org/>
PREFIX mydict: <http://mydictionary.org/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX sumo:
<http://www.ontologyportal.org/translations/SUMO.owl.txt#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT ?origLabel ?origLanguage ?origNotes
?origAltSpelling ?origEType
?eqLabel ?eqLanguage ?eqNotes ?eqAltSpelling ?eqEType
?partOfSpeech ?field ?context ?example
WHERE      { ?orig rdfs:label ?origLabel ;
dc:language ?origLanguage ;
rdf:type ?origEType ;
ex:hasEquivalent ?eq .

?eq rdf:Value ?eqValue .
?eq ex:hasEquivalencePOS ?partOfSpeech .
?eq ex:hasEquivalenceField ?field .
?eqValue rdfs:label ?eqLabel ;
      dc:language ?eqLanguage ;
      rdf:type ?eqEType .
?orig dc:language "EN" .
{ ?eqValue dc:language "EN" . }
UNION { ?eqValue dc:language "ES" . }
OPTIONAL { ?orig skos:note ?origNotes . }
OPTIONAL { ?orig skos:altSpelling ?origAltSpelling . }
OPTIONAL { ?eqValue skos:altSpelling ?eqAltSpelling . }
OPTIONAL { ?eqValue skos:note ?eqNotes . }
OPTIONAL { ?eq ex:hasEquivalenceContext ?context . }
OPTIONAL { ?eq ex:hasEquivalenceExample ?example . }
FILTER regex(?origLabel, "so","i" )
FILTER regex(?origLabel, "far","i" )
FILTER regex(?origLabel, "so","i" )
FILTER regex(?origLabel, "good","i" )
}
ORDER BY ?origLabel ?eqLanguage ?partOfSpeech ?eqLabel
```

11.2 Apéndice B: Manual de usuario

La aplicación CatSubs (Computer Aided Translation of Subtitles) es una herramienta que facilita la traducción de subtítulos, integrando la consulta de un repositorio terminológico actualizable en todo momento por el propio traductor.

El repositorio o diccionario puede modificarse y consultarse de forma independiente o desde la misma ventana de traducción.

La ventana de traducción permite mantener a la vista tanto los subtítulos en la lengua de origen (inglés en la versión actual) como los subtítulos en la lengua meta (castellano en la versión actual), y visualizar una cualquiera de ambas series de subtítulos sincronizadas con el vídeo correspondiente.

Tanto las ventanas de consulta como las de edición y subtitulación pueden mantenerse abiertas a la vez en el programa, y el usuario puede alternar entre ellas según le convenga a través de las pestañas correspondientes, o bien visualizar varias simultáneamente. Se accederá a todas las ventanas desde la pantalla principal de la aplicación.

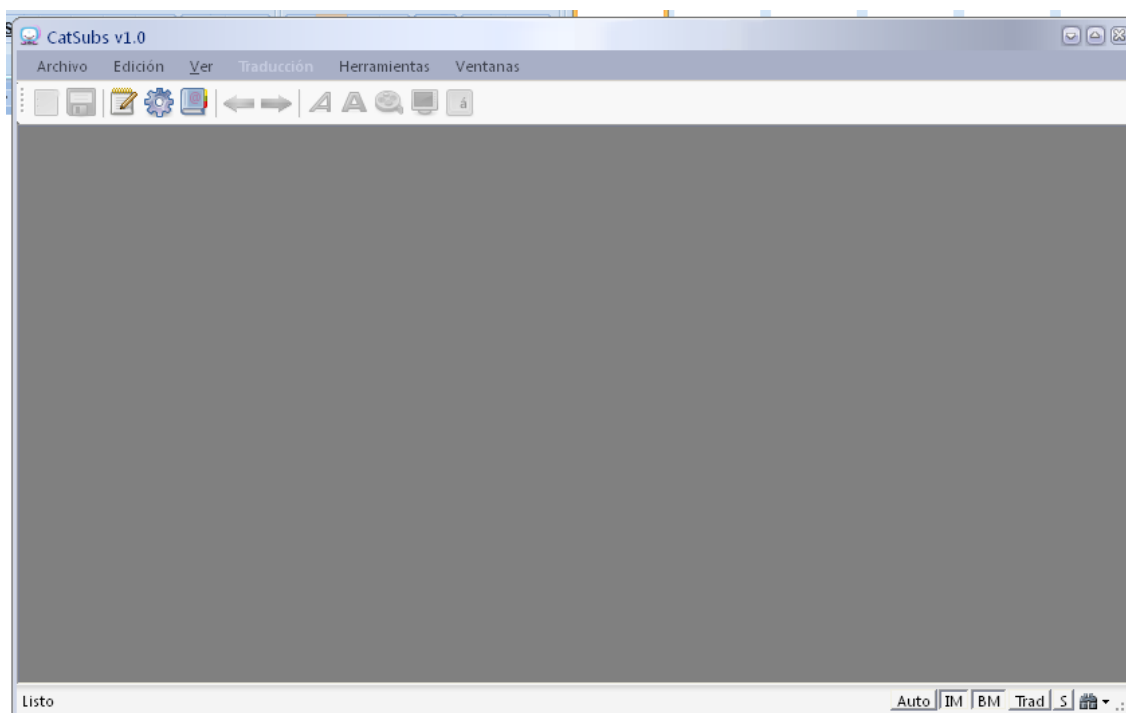


Ilustración 35: Pantalla principal





























	A	B	C	D	E	F
1						
2						
3						
4						
5						

Tabla 12: Guía de referencia - botones

11.2.1 Edición del diccionario

El diccionario puede modificarse desde la ventana destinada a tal efecto. Para abrir la ventana, puede pulsar en **Menú Archivo -> Editar diccionario (Alt + M)** o en el botón de edición de diccionario (*D1* en la guía de referencia de botones). Para no sobrecargar el sistema, la aplicación permite abrir un máximo de tres ventanas de edición.

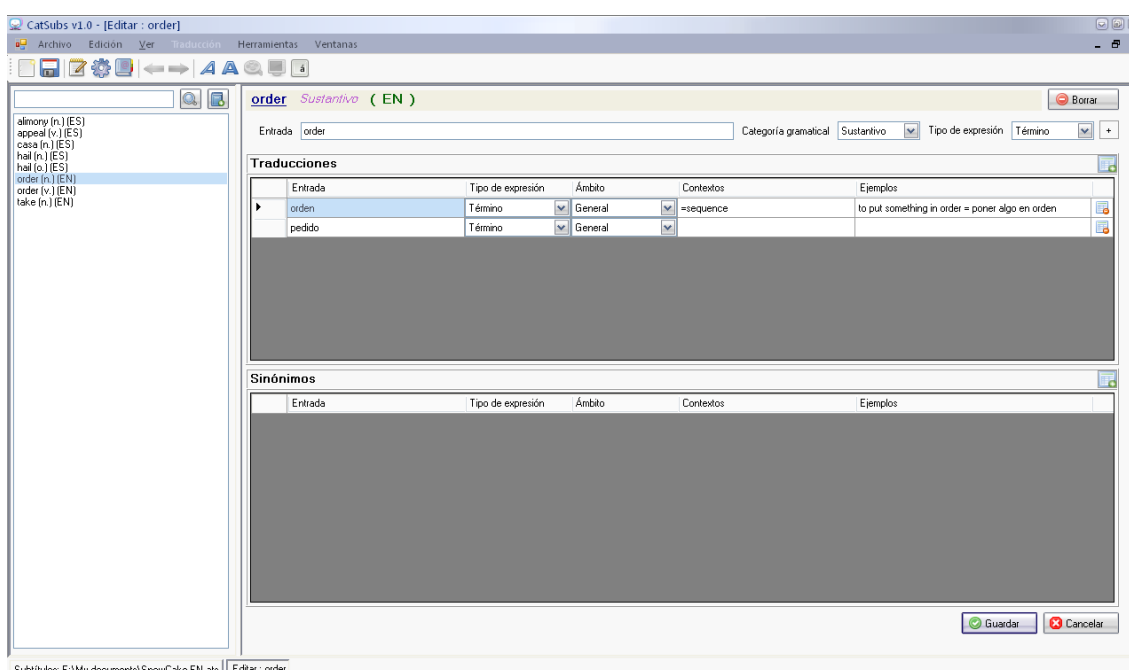


Ilustración 36: Pantalla de edición de diccionario


11.2.1.1 Nueva entrada

Para introducir una nueva expresión en el diccionario, basta con pulsar bien en el botón *B2*, en el botón *A1* de la barra de herramientas, o en **Menú Archivo -> Nuevo (Ctrl + N)**. Entonces se mostrará el panel de edición, donde podrá rellenar todos los campos correspondientes a la nueva entrada:

- **Entrada:** representación gráfica de la entrada.
- **Categoría gramatical:** sustantivo, verbo, adjetivo, adverbio y otros.
- **Tipo de expresión:** término, locución, modismo y forma compuesta.
- **Grafía alternativa:** si se puede representar de dos formas distintas. Por ejemplo, “periodo” o “período”, “psicología” o “sicología”.
- **Idioma:** lengua de la entrada.
- **Observaciones:** comentarios o anotaciones relevantes sobre la entrada.
- **Tabla de traducciones:** se podrán introducir tantas traducciones como se desee (entre el idioma de la entrada y el otro idioma de trabajo) pulsando en *D2*, de las que se podrá incluir la expresión, el tipo de expresión, el ámbito de aplicación (seleccionándolo de una lista desplegable), y opcionalmente, contextos y ejemplos para facilitar posteriores consultas. Para borrar una traducción, debe pulsarse el botón *E2*.
- **Tabla de sinónimos:** tiene el mismo funcionamiento que la tabla de traducciones. La introducción de contextos en ambas tablas puede realizarse de dos maneras. La primera, introduciéndolos directamente en la casilla, separándolos mediante punto y coma, y opcionalmente anteponiendo “+” si el contexto representa un objeto, o “=” si representa una equivalencia. Si lo prefiere, también puede hacer doble clic en la casilla correspondiente para abrir una ventana de introducción de contextos.


Igualmente, puede introducir los ejemplos directamente o mediante una nueva ventana, haciendo doble clic sobre la casilla.

11.2.1.2 *Modificar entrada*

Introduciendo un término o expresión en la casilla de búsqueda y pulsando en el botón *C2*, se realizará la consulta en el diccionario, en los dos idiomas de trabajo. La aplicación le mostrará a la izquierda una lista con las coincidencias en ambos idiomas, y podrá seleccionar cualquiera de ellas. La entrada seleccionada se mostrará en el panel de edición, donde podrá editar los campos que desee. Para guardar los cambios, pulse en , en el botón *B1* de la barra de herramientas, o seleccione **Menú archivo -> Guardar**.

Si cierra el formulario sin haber guardado la entrada creada o modifica, el programa se lo indicará para que pueda guardarla, ignorar los cambios o cancelar el cierre de formulario.

11.2.1.3 *Borrar entrada*

Para borrar una entrada nueva o una ya existente en el diccionario, basta con pulsar en el botón  del panel de edición.

11.2.1.4 *Navegación entre últimas consultas*

Puede navegar entre las cinco últimas consultas realizadas pulsando en los botones *F1* y *A2* de la barra de herramientas.

11.2.2 Consulta del diccionario

Para no reducir su uso exclusivamente al contexto de la traducción audiovisual, el diccionario puede consultarse de forma independiente, sin necesidad de tener cargado ningún archivo de subtítulos. Esto se lleva a cabo desde la ventana “Consulta del diccionario”. Para abrirla, debe seleccionarse **Menú Archivo -> Consultar diccionario (Alt + D)** o bien pulsar en el botón *E1* de la barra de herramientas.

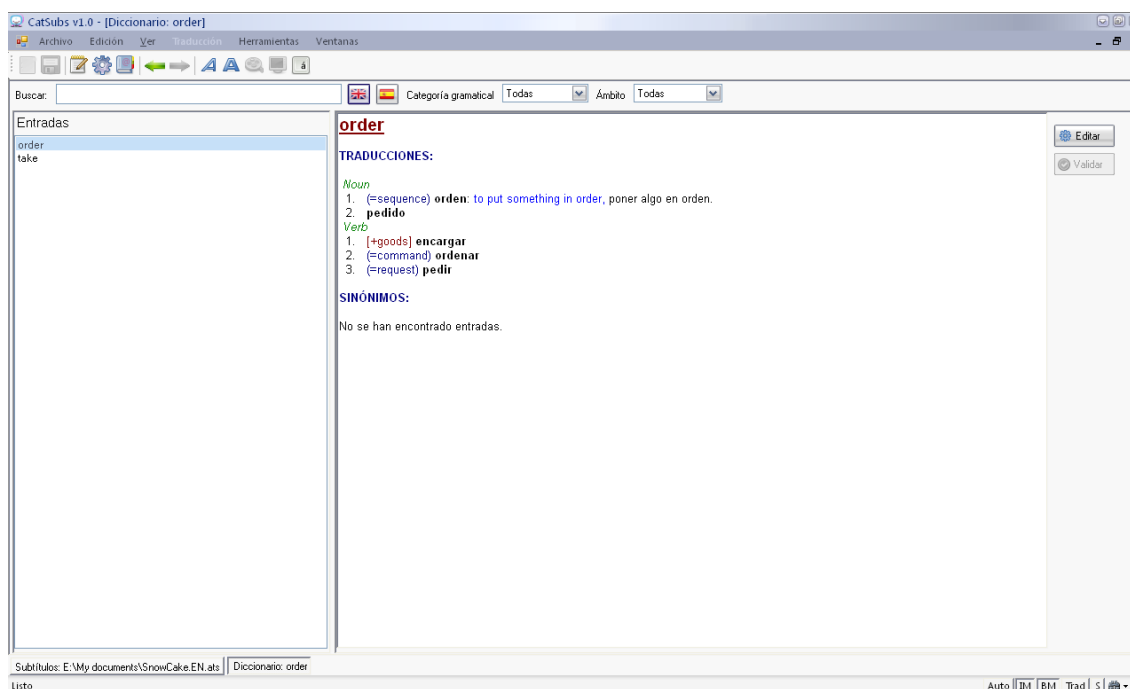


Ilustración 37: Ventana de consulta del diccionario

En esta ventana se puede realizar la búsqueda tanto en la lengua de origen como en la lengua meta, pulsando en el botón correspondiente después de introducir la cadena a buscar. Tiene la opción de limitar la búsqueda a una categoría gramatical determinada, así como a un ámbito concreto (General, Administración, Economía, Ingeniería...).

Una vez efectuada la búsqueda, se devolverá una lista con todas las coincidencias, y la primera de las entradas se mostrará en el panel central. Para visualizar otra entrada, selecciónela en la lista de Entradas. Si desea modificar la entrada seleccionada, pulse en el botón “Editar” y se cargará la entrada en un formulario de edición de diccionario, ya descrito en el apartado anterior. Al igual que en la ventana de edición, puede navegar entre las cinco últimas consultas realizadas pulsando en los botones *F1* y *A2*.

11.2.2.1 *Búsqueda no segura*

La aplicación ofrece la posibilidad de realizar Búsquedas no seguras. Esta opción consiste en la búsqueda en el repositorio de términos y expresiones relacionados entre sí, pero no directamente. De esta forma, si por ejemplo el diccionario recoge la entrada “cinta” en castellano y para ésta figura la traducción “ribbon” en inglés, que a su vez tiene la traducción “lazo” al castellano, al buscar sinónimos de “cinta” se le devolverá el resultado no seguro “lazo”. Usted tiene la opción de validar esa relación, en cuyo caso se añadirá al diccionario de forma que en la próxima búsqueda segura (modo de trabajo recomendado), se le mostrará el sinónimo como tal. Para validar una entrada no segura, selecciónela en el panel y pulse en el botón “Validar”. El programa le indicará el éxito de la operación mostrando el icono junto al botón. Si no se puede validar la entrada seleccionada, se lo indicará con un mensaje.

11.2.3 Traducción de subtítulos

El tercer y último tipo de ventana primaria es la de traducción de subtítulos. Para abrirla, seleccione **Menú Archivo -> Abrir ventana de subtitulación** o pulse en el botón *C1* de la barra de herramientas. Se abrirá una ventana de subtitulación vacía, con un panel de sugerencias a la derecha.

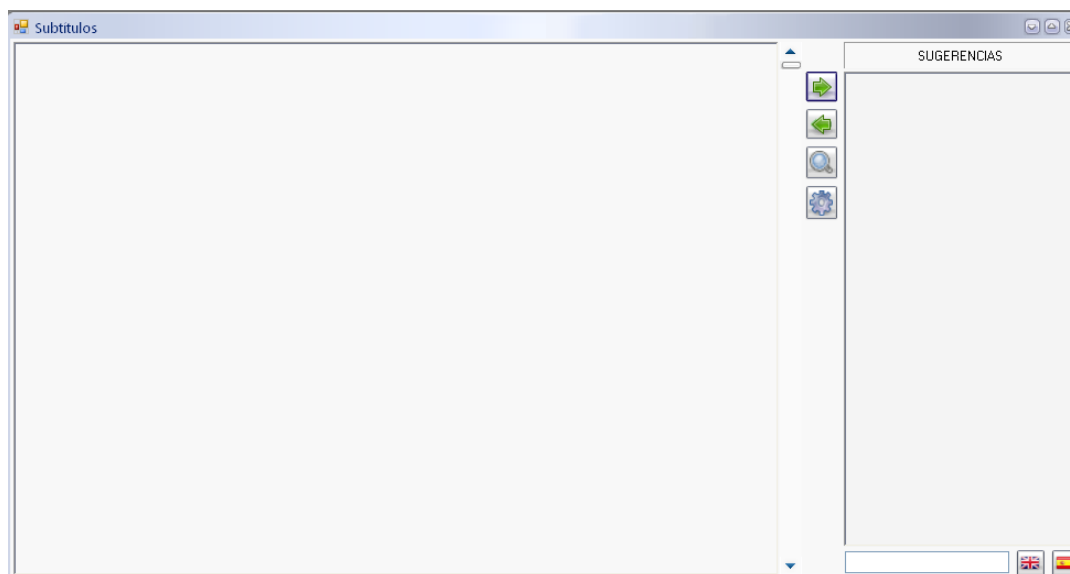


Ilustración 38: Ventana de subtitulación – sin subtítulos

11.2.3.1 Cargar archivos de subtitulación

En todo momento puede tener cargados dos archivos: el que contiene los subtítulos en la lengua de origen (fuente), y el que contiene los subtítulos en la lengua meta (destino). Ambas columnas de subtítulos se mostrarán alineadas. Opcionalmente, puede mantener abierto solo uno de los dos archivos.

Los archivos de subtitulación que utiliza el programa tienen el formato propio .ats (Assisted Translation Subtitles).

Para cargar un archivo de subtítulos en el idioma original, seleccione **Menú Traducción -> Archivo fuente -> Abrir (Ctrl + K)**, y seleccione un archivo .ats en la lengua de origen. Se mostrarán los subtítulos en el panel principal.

Para cargar un archivo de subtítulos en la lengua meta, seleccione **Menú Traducción -> Archivo destino -> Abrir (Ctrl + L)** y seleccione un archivo .ats en la lengua meta. Se mostrarán los subtítulos en el panel principal, alineados con los subtítulos de origen si están cargados. Si desea crear un archivo destino nuevo a partir del archivo fuente (manteniendo numeración y códigos de tiempo), seleccione **Menú Traducción -> Archivo destino -> Nuevo (a partir de archivo fuente)**, o bien seleccione **Menú Archivo -> Nuevo (Ctrl + N)** o pulse en el botón A1 de la barra de herramientas.

Una vez cargado un archivo, puede acceder al menú correspondiente pulsando con el botón derecho sobre la cabecera, además de la vía habitual desde la barra de menú.

Puede navegar entre todos los subtítulos empleando la barra de desplazamiento que se encuentra a la derecha del panel. Para ir a un subtítulo determinado, seleccione **Traducción -> Ir a -> Subtítulo...** (**Ctrl + Mayús. + S**) o pulse en el botón **F3** de la barra de herramientas. Aparecerá una ventana de selección de subtítulo. También puede navegar entre los cinco últimos subtítulos seleccionados mediante los botones **F1** y **A2** de la barra de herramientas.

11.2.3.2 *Editar subtítulos*

Puede modificar cualquier subtítulo situando el ratón sobre él. El texto puede ponerse en cursiva, pulsando en el botón **B3** de la barra de herramientas, o en negrita, pulsando en el botón **C3**. Cualquier texto seleccionado puede marcarse en cursiva, mientras que la negrita se aplica al conjunto del subtítulo.

Para deshacer o rehacer cambios en el subtítulo actual, seleccione **Menú Edición -> Deshacer (Ctrl + Z)** o **Menú Edición -> Rehacer (Ctrl + Y)**, respectivamente. Igualmente, puede cortar, copiar, pegar o seleccionar todo el texto desde el mismo menú, o mediante las teclas de acceso rápido **Ctrl + X**, **Ctrl + C**, **Ctrl + V** o **Ctrl + A**, respectivamente.

11.2.3.3 *Información de los subtítulos*

En cada subtítulo se muestra el número de subtítulo, los códigos de tiempo de entrada y salida y la duración, además de un indicador de longitud relativa. Dicha cifra indica el número de caracteres que por exceso (en rojo) o por defecto (en verde) distan de la longitud ideal máxima del subtítulo, para la duración de éste. El programa lo calcula a partir de la regla de los seis segundos: en un período de seis segundos el espectador es capaz de leer y asimilar la información contenida en dos líneas de subtítulos, cuando cada línea contiene un máximo de 35 espacios o pulsaciones.

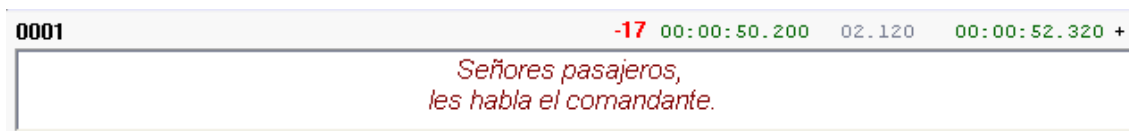


Ilustración 39: Subtítulo

Si alguna línea del subtítulo supera los caracteres máximos indicados por el usuario en la Configuración (véase apartado 11.3.4.2, *Configuración*), el subtítulo se resaltará mediante un borde rojo.

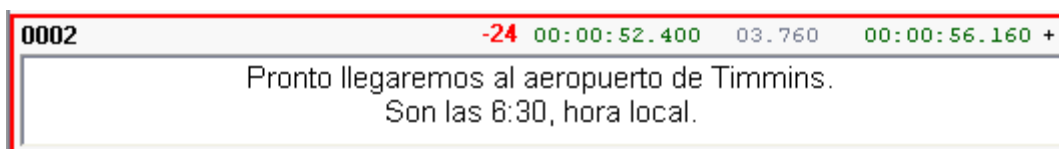


Ilustración 40: Subtítulo con indicación de longitud excesiva

Pulsando sobre el número de subtítulo, se introducirá una marca ámbar, que le servirá para revisar más adelante los subtítulos marcados (*véase apartado 11.3.3.4, Ir a subtítulos determinados*).

Finalmente, en la esquina superior derecha del subtítulo se muestra un símbolo “+”. Pulsando sobre este símbolo, se abrirá una ventana en la que podrá introducir comentarios o anotaciones sobre el subtítulo actual (aclaraciones, dudas, etcétera) que serán de utilidad en caso de que varias personas trabajen con el mismo archivo (traductores, adaptadores, correctores...). Si un subtítulo contiene anotaciones, el símbolo “+” se mostrará sobre un fondo azul, y pasando por encima de este con el ratón, se hará visible la anotación correspondiente.



Ilustración 41: Marca y anotación en subtítulo

11.2.3.4 Ir a subtítulos determinados

La aplicación ofrece la posibilidad de saltar directamente a los subtítulos que cumplan una característica determinada, mediante **Menú Traducción -> Ir a**. Las opciones son:

- **Marca (Ctrl + Mayús. + M):** salta entre todos los subtítulos que han sido marcados previamente.
- **Observaciones (Ctrl + Mayús. + N):** salta entre los subtítulos que contienen anotaciones.
- **Subtítulos vacíos (Ctrl + Mayús. + E):** salta entre los subtítulos que aún no han sido traducidos.
- **Subtítulo... (Ctrl + Mayús. + S):** muestra la ventana de selección de subtítulo.

11.2.3.5 Guardar y cerrar archivos de subtítulos

Para guardar el archivo activo de subtítulos, seleccione **Menú Archivo -> Guardar (Ctrl + S)** desde la ventana de subtitulación o pulse en el botón de la barra de herramientas **B1**. Opcionalmente, para guardar el archivo fuente, seleccione **Menú**

Traducción -> Archivo fuente -> Guardar (Ctrl + 1) o Traducción -> Archivo fuente -> Guardar como (Ctrl + Mayús. + 1). Para guardar el archivo de destino, seleccione **Menú Traducción -> Archivo destino -> Guardar (Ctrl + 2) o Traducción -> Archivo destino -> Guardar como (Ctrl + Mayús + 2).**

Para cerrar un archivo, vaya a **Menú Traducción -> Archivo [tipo] -> Cerrar**, siendo [tipo] “fuente” o “destino”, en función del archivo que desee cerrar. Si el archivo no ha sido guardado desde la última modificación, se le informará de ello y se le ofrecerá la posibilidad de guardarlo, ignorar los cambios o cancelar el cierre del archivo.

11.2.3.6 Importar y exportar subtítulos a otros formatos

En su versión actual, la aplicación permite importar y exportar a formato SubRip (.srt). Para ello, seleccione **Menú Traducción -> Archivo [tipo] -> Importar o Menú Traducción -> Archivo [tipo] -> Exportar.**

11.2.3.7 Cargar vídeo

Para cargar un archivo de vídeo sobre el que visualizar los subtítulos (formatos admitidos: .avi, .mpeg, .mpg, .wmv, .asf), seleccione **Menú Archivo -> Abrir película (Ctrl + O)**, o pulse en el botón *D3* de la barra de herramientas. Una vez seleccionado un archivo de vídeo, se mostrará en una pantalla integrada en la ventana de subtitulación. Puede mostrar dicha pantalla u ocultarla cuando no la necesite seleccionando **Archivo Ver -> Ver vídeo**, o pulsando en el botón *E3* de la barra de herramientas.

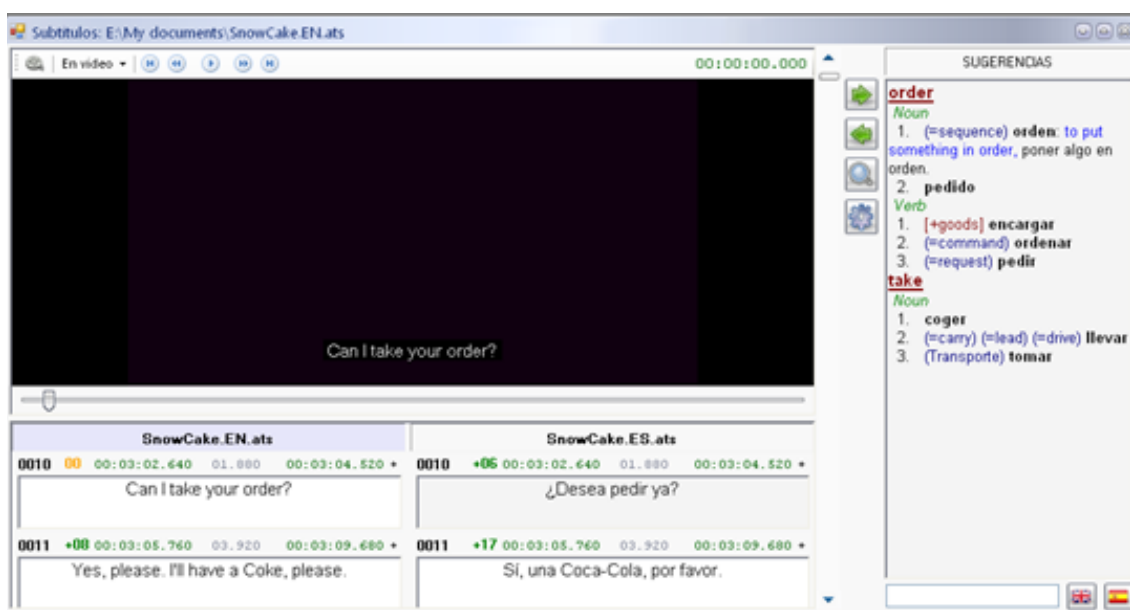


Ilustración 42: Ventana de subtitulación

11.2.3.8 Controles de vídeo

La pantalla de visualización del vídeo incluye una serie de controles que permiten realizar diversas acciones, tal como se muestra en la tabla 13.







Botón	Función
	Cargar otro archivo de vídeo
	Rebobinar al subtítulo anterior
	Rebobinar un segundo
	Reproducción / pausa
	Adelantar al subtítulo siguiente
	Adelantar un segundo

Tabla 13: Controles de vídeo

Además, se puede emplear la barra de desplazamiento inferior para desplazarse por la imagen. La posición temporal del fotograma actual se indica en la esquina superior derecha de la pantalla.

11.2.3.9 Subtítulos activos

Si hay abierto algún archivo de subtítulos, en cada momento se mostrará en pantalla el subtítulo actual, superpuesto sobre el vídeo. Para cambiar los subtítulos que se muestran en pantalla (subtítulos activos), pulse en el menú desplegable **En vídeo**, y seleccione **Fuente** o **Destino**. También puede cambiarlos haciendo clic con el botón izquierdo del ratón sobre la cabecera de la columna de los subtítulos que desee activar. La columna activa se distingue por el color azul de la cabecera.

11.2.3.10 Autobúsqueda en el diccionario

Hay varias formas de consultar el diccionario desde la ventana de subtitulación. Se puede activar la opción de “Autobúsqueda de subtítulo”, bien pulsando en el botón A5 en la barra de estado, bien modificando las preferencias (véase apartado 11.3.4.2, Configuración). En caso de que la opción esté activada, cada vez que se entre en un subtítulo, la aplicación procederá a la búsqueda de todo el contenido del subtítulo en el diccionario mientras usted continúa trabajando, y mostrará los resultados en el panel de “Sugerencias”, situado a la derecha.

Dependiendo del modo de trabajo seleccionado, se realizará la búsqueda en un idioma o en otro. Así, si se está en modo Traducción y con los dos archivos abiertos, al entrar en un subtítulo de la columna destino, se realizará la consulta del texto del subtítulo equivalente en la columna fuente, de forma que se buscará de la lengua original a la lengua meta. Si, por el contrario, se está en modo Corrección, al entrar en

un subtítulo de la columna destino, se buscará el texto del mismo subtítulo, realizándose la consulta de sinónimos en el diccionario.

Para cambiar el modo, puede pulsar el botón *D5* en la barra de estado, o acceder desde el formulario de Configuración (véase *apartado 11.3.4.2, Configuración*).

11.2.3.11 *Búsqueda manual en el diccionario*

También se pueden realizar búsquedas en el diccionario de forma manual. Una forma es introducir la cadena a buscar en la casilla de búsqueda situada en la esquina inferior derecha de la ventana, y pulsar en el botón del idioma correspondiente (*E4* o *F4*).

Otra forma es seleccionar el texto en cualquiera de los subtítulos y pulsar en el botón *C4*, situado a la izquierda del panel de sugerencias. De esta forma, dependiendo del idioma del subtítulo y del modo de trabajo, se realizará la búsqueda de traducciones o sinónimos en el idioma adecuado, y se mostrará en el panel de sugerencias. No obstante, si se activa la opción Búsqueda automática, pulsando en el botón *C5* de la barra de estado, o mediante el formulario de Configuración (véase *apartado 11.3.4.2, Configuración*), bastará con seleccionar un fragmento de texto para que se proceda a su búsqueda en el repositorio.

11.2.3.12 *Inserción de sugerencias en el subtítulo*

Para introducir una entrada de la ventana de sugerencias en el subtítulo actual, seleccione el texto y pulse en el botón *B4*, a la izquierda del panel de sugerencias. Si desea introducir el texto solo con seleccionarlo, puede activar la opción Inserción automática, pulsando el botón *B5* en la barra de estado, o mediante el formulario de Configuración (véase *apartado 11.3.4.2, Configuración*).

11.2.3.13 *Visualización extendida de sugerencias*

Las sugerencias mostradas en el panel lateral pueden visualizarse en modo extendido, que muestra, además de la categoría gramatical, los ámbitos, contextos y ejemplos correspondientes a cada relación de traducción o sinonimia.

Si desea contraer las sugerencias para que no se muestren ámbitos, contextos ni ejemplos, pulse en la cabecera del panel Sugerencias, o bien desmarque **Menú Traducción -> Visualización extendida de sugerencias (F2)**.

11.2.3.14 *Modificación del diccionario*

Si desea modificar alguna entrada sugerida, pulse en el botón **D4**, situado a la izquierda del panel de sugerencias. La aplicación abrirá una ventana de edición del diccionario donde podrá editar las entradas mostradas.

Si desea añadir una nueva entrada a partir de una porción de texto de los subtítulos, seleccione el texto y pulse en el botón **A4**. De nuevo, se abrirá una ventana de edición donde podrá añadir campos a la entrada y guardarla en el diccionario.

11.2.3.15 *Extracción por lotes de términos y expresiones*

Para evitar interrumpir el proceso de traducción con constantes inserciones y ediciones del diccionario, CatSubs ofrece la posibilidad de realizar una extracción por lotes de términos y expresiones seleccionados en la ventana de subtitulación, para su introducción en el repositorio.

Durante la traducción o revisión, inserte una marca en los subtítulos que contienen texto que desea añadir al diccionario. Posteriormente, una vez finalizada la traducción (o en el momento que desee), puede abrir un formulario de extracción por lotes, seleccionando **Menú Traducción -> Extracción por lotes (F4)**.

Este formulario le mostrará en diversas filas el texto (en minúsculas para mayor comodidad) de todos los subtítulos marcados, así como una serie de campos a rellenar. Si hay abiertos tanto archivo de origen como de destino, se incluirá el texto de ambos pares de subtítulos.

Cuando haya realizado las modificaciones necesarias, pulse en el botón Guardar y se almacenarán las nuevas entradas en el diccionario. Si alguna entrada ya existiese, simplemente se añadirá la información adicional.

11.2.3.16 *Estadísticas de los subtítulos actuales*

Se pueden consultar diversas estadísticas sobre los subtítulos activos seleccionando **Herramientas -> Estadísticas**. Se mostrará un formulario con información de interés para el traductor, como cantidad de subtítulos, de palabras, de caracteres, número de subtítulos con marcas o anotaciones, o cantidad de subtítulos vacíos.

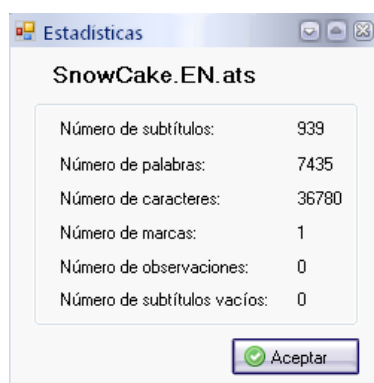


Ilustración 43: Estadísticas

11.2.4 Otras utilidades

11.2.4.1 Cambio de idioma de la interfaz gráfica de usuario

Si lo desea, puede cambiar el idioma de la interfaz de usuario a otro de los idiomas de trabajo. Actualmente, se encuentran disponibles el inglés y el español. Para realizar el cambio, seleccione **Menú Herramientas -> Idioma de la interfaz de usuario**. Se le mostrará una ventana con una lista desplegable de idiomas. Seleccione el idioma deseado, pulse en el botón Aceptar y reinicie la aplicación si desea que se apliquen los cambios.

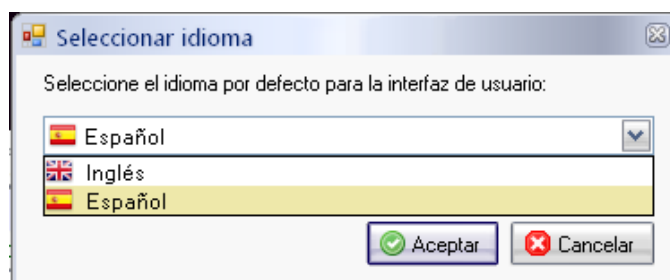


Ilustración 44: Formulario de cambio de idioma

11.2.4.2 Configuración

El programa permite la modificación por parte del usuario de varias opciones de configuración, que se almacenarán de una sesión a otra. Para acceder al formulario de configuración, seleccione **Menú Herramientas -> Configuración**.

La ventana de configuración se divide en dos pestañas: Traducción y Diccionario.

OPCIONES DE TRADUCCIÓN

En la pestaña de traducción se pueden modificar las siguientes opciones:

- **Permitir hasta 4 pares de subtítulos simultáneamente:** Si el sistema dispone de suficiente memoria, puede activar esta opción y permitir así abrir hasta cuatro ventanas de subtitulación. Esto puede ser útil si se trabaja a la vez con varios archivos relacionados entre sí (dos capítulos de una misma serie, o contenido extra de un mismo DVD), o simplemente si se está trabajando en varias traducciones.
- **Autobúsqueda de subtítulo:** Permite buscar automáticamente en el diccionario el contenido del subtítulo cuando el usuario entre en él, mientras escribe.
- **Corrección automática mayúsculas/minúsculas:** Para corregir la introducción accidental de dos mayúsculas seguidas de minúsculas.
- **Número máximo de caracteres por subtítulo:** Puede ajustar el número máximo de caracteres que deben permitirse en un subtítulo, en función de las exigencias del cliente. Por defecto, este valor es de 42.
- **Modo de trabajo:** Si se selecciona el modo “Traducción”, se realizará la autobúsqueda del contenido del subtítulo en la lengua original. Si por el contrario se trabaja en modo “Corrección”, se buscará el contenido del subtítulo en la lengua meta.
- **Búsqueda de texto:**
 - **Con selección:** se realiza la búsqueda automática del texto seleccionado en el diccionario.
 - **Pulsando botón:** Se realiza la búsqueda cuando el usuario pulse el botón correspondiente, *C4*.
- **Inserción de término:**
 - **Con selección:** Se inserta automáticamente en el subtítulo actual el término seleccionado en el panel de sugerencias.
 - **Pulsando flecha:** Se inserta el texto seleccionado desde el panel de sugerencias cuando el usuario pulse el botón correspondiente, *B4*.

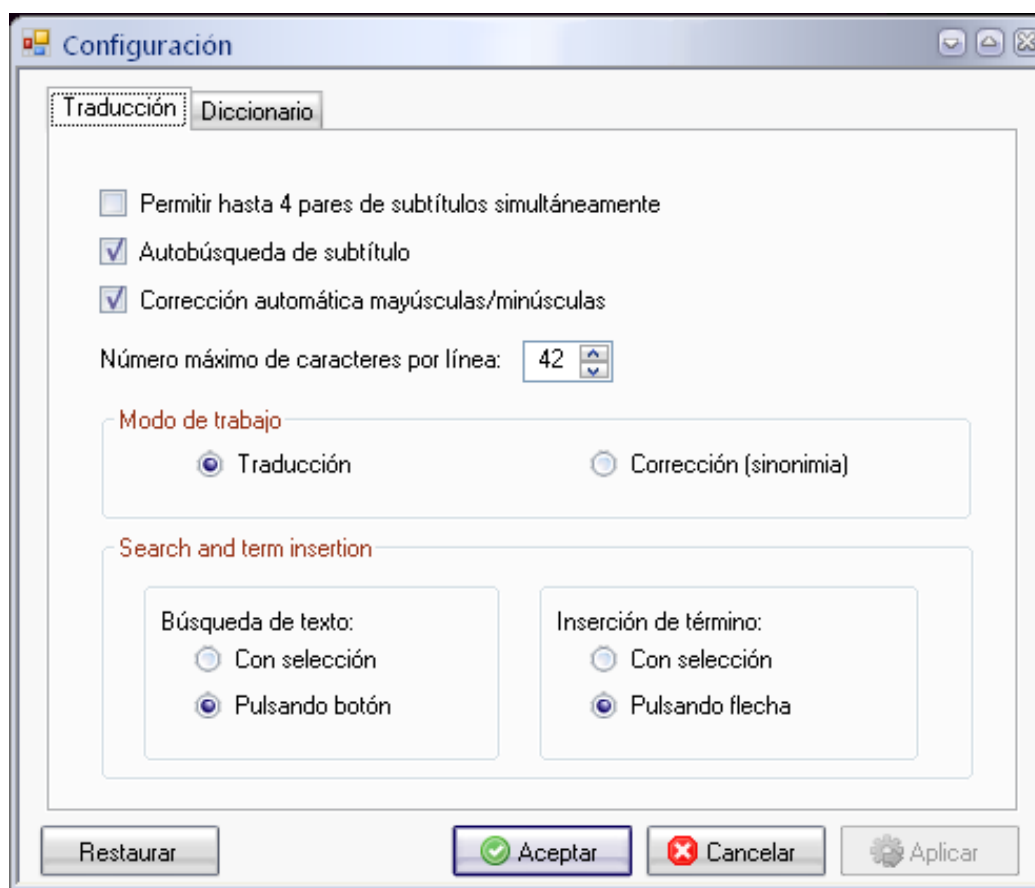


Ilustración 45: Formulario de configuración – opciones de traducción

OPCIONES DE DICCIONARIO

Las opciones que se pueden modificar desde la pestaña de diccionario se enumeran a continuación:

- **Abrir diccionario al iniciar el programa:** Pulse esta opción si desea que el diccionario se cargue siempre que se inicie el programa.
- **Almacenar diccionario en un archivo:** Por defecto, el repositorio se almacena en una base de datos MySQL. Para ello, deberá tener una base de datos de nombre “wordrepo” y crear un usuario de nombre “semweb” y contraseña “semweb”. Si no dispone de esta opción, active esta casilla para que el diccionario se almacene en un archivo, aunque el acceso a este será más lento cuando alcance un tamaño considerable.
- **Opciones de búsqueda**

Opciones que se aplicarán en la ventana de consulta del diccionario:

- **Búsqueda no segura:** Si marca esta casilla, cuando se consulte el diccionario se realizará una búsqueda no segura, mostrando resultados

relacionados entre sí por medio de otros resultados intermedios. También puede activarse desde el botón *E5* de la barra de estado.

- **Profundidad de búsqueda:** Si activa la búsqueda no segura, puede indicar la profundidad de búsqueda, es decir, el número máximo de elementos intermedios que puede haber entre el término buscado y el resultado no seguro. Puesto que en una base de datos grande puede ralentizarse bastante la búsqueda, se limita a un máximo de cinco elementos. Si se producen búsquedas muy lentas, se recomienda reducir la profundidad de búsqueda o desactivar la opción de búsqueda no segura.
- **Recordar opciones de búsqueda:** Márquela si desea que se recuerde el ámbito y la categoría gramatical a buscar en la ventana de consulta del diccionario.
- **Número máximo de resultados:** El número máximo de entradas que se devolverán del diccionario tras una consulta.
- **Buscar:**
 - Todo: Se buscan en el diccionario las entradas que contengan todas las palabras de la cadena de búsqueda.
 - Frase exacta: Se busca exactamente la cadena de búsqueda especificada.
 - Cualquier palabra: Se busca cualquier palabra contenida en la cadena de búsqueda.
- **Opciones de edición**

Opciones que se aplicarán en la ventana de edición del diccionario.

 - **Mostrar:**
 - Traducciones y sinónimos: Se muestran tanto las traducciones como los sinónimos correspondientes a esa entrada.
 - Solo traducciones.
 - Solo sinónimos.
 - **Categoría gramatical por defecto:** Categoría gramatical que se incluirá por defecto en las nuevas entradas.
 - **Ámbito por defecto:** Ámbito que se incluirá por defecto en las nuevas entradas.

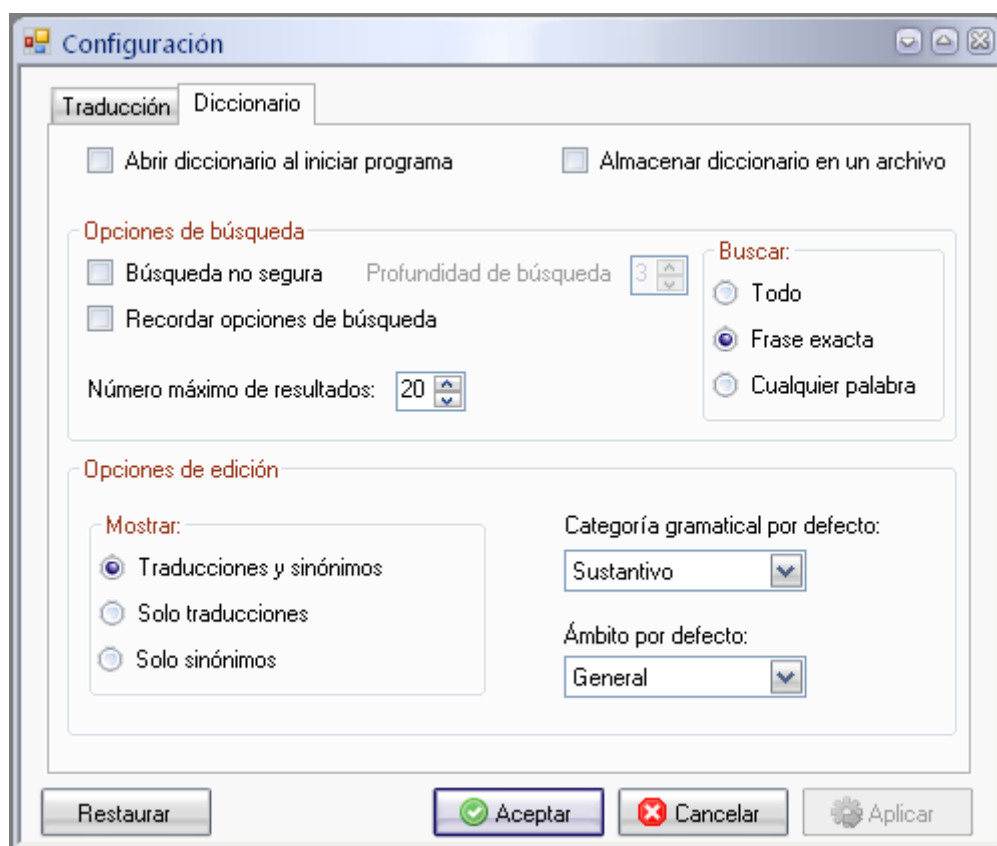


Ilustración 46: Formulario de configuración - opciones de diccionario